WYDANIE
**AKTUALNE**

# INFORMATYKA

PODRĘCZNIK • LICEUM I TECHNIKUM • ZAKRES PODSTAWOWY

**NOWA**
**EDYCJA**

1

WSiP

Autorzy: **Wanda Jochemczyk, Katarzyna Olędzka**
Podręcznik dopuszczony do użytku szkolnego przez ministra właściwego do spraw oświaty i wychowania i wpisany do wykazu podręczników przeznaczonych do kształcenia ogólnego do nauczania informatyki, na podstawie opinii rzeczoznawców: **prof. dr hab. Liliany Byczkowskiej-Lipińskiej**, **dr. inż. Łukasza Tomczyka**, **prof. dr. hab. Marka Ruszkowskiego**.

Etap edukacyjny: III
Typ szkoły: szkoła ponadpodstawowa (liceum ogólnokształcące i technikum)
Zakres: podstawowy
Rok dopuszczenia: 2022

Numer ewidencyjny w wykazie: 974/1/2022/z1

Podręcznik wpisany do wykazu podręczników MEiN dopuszczonych do użytku szkolnego, uwzględniających podstawę programową kształcenia ogólnego określoną w rozporządzeniu z dnia 30 stycznia 2018 (Dz. U. poz. 467 oraz z 2020 r. poz. 1248 i z 2021 r., poz. 1537).

© Copyright by Wydawnictwa Szkolne i Pedagogiczne
Warszawa 2019, 2022

Wydanie II zmienione, 1/2025

ISBN 978-83-02-20402-9 – podręcznik w wersji drukowanej
ISBN 978-83-02-22930-5 – podręcznik w wersji elektronicznej (flipbook)

Opracowanie merytoryczne i redakcyjne: **Beata Brzeg-Wieluńska** (redaktor koordynator),
**Maria Białek** (redaktor merytoryczny)
Konsultacja naukowa: **dr Leszek Rudak, Romualda Laskowska**
Redakcja językowa: **dr Tomasz Karpowicz**
Korekta językowa: **Jolanta Kucharska**
Redakcja techniczna: **Janina Soboń**
Projekt okładki: **Marta Staniszewska**
Projekt graficzny: **Hanna Michalska-Baran**
Opracowanie graficzne: **Image Niokoba LTD sp. z o.o.**
Fotoedycja: **Marta Bażyńska**
Skład i łamanie: **Piotr Uszyński**

Wydawnictwa Szkolne i Pedagogiczne Spółka Akcyjna
00-807 Warszawa, Aleje Jerozolimskie 96
KRS: 0000595068
Infolinia: 801 220 555
**www.wsip.pl**

# Spis treści


Infografika: Informatyka jest wszędzie ..... 5
Jak korzystać z podręcznika ..... 6

**Ochrona danych i oprogramowania** ..... 7
Infografika: Ochrona danych osobowych w sieci ..... 8
1. Bezpieczna praca z komputerem ..... 10

**Arkusz kalkulacyjny** ..... 23
Infografika: Zastosowania arkusza kalkulacyjnego ..... 24
2. Podstawy pracy z arkuszem ..... 26
3. Stosowanie instrukcji warunkowych ..... 40
4. Porównywanie ofert w arkuszu ..... 49
5. Zakupy grupowe z arkuszem ..... 56

**Grafika rastrowa** ..... 63
Infografika: Zdjęcia – retusz kolorów ..... 64
6. Podstawy edycji grafiki rastrowej ..... 66
7. Praca na warstwach ..... 76
8. Edycja fotografii ..... 85
Infografika: Sposób na dobre selfie ..... 100
9. Projektowanie okładki książki i e-booka ..... 101

**Edytor tekstu** ..... 105
Infografika: Sposób zapisu liczb i tekstu ..... 106
10. Podstawy edycji tekstu ..... 108
11. Przygotowanie publikacji do druku ..... 119
12. Dokumenty o złożonej strukturze ..... 132
13. Korespondencja seryjna ..... 142

**Algorytmika i programowanie w Pythonie*** ..... 151
Infografika: Podstawowe konstrukcje programistyczne ..... 152
14. Podstawy pracy w środowisku Python ..... 154
15. Definiowanie funkcji obliczeniowych ..... 166
16. Wyszukiwanie wzorca w tekście ..... 176
Infografika: Algorytm ..... 184
17. Przetwarzanie napisów ..... 188
Infografika: Schematy blokowe ..... 193
Infografika: ⌜·○>⌞·⌜>, czyli szyfry ..... 198
18. Szyfrowanie i deszyfrowanie tekstu ..... 200

**Algorytmika i programowanie w C++*** ..... 207
Infografika: Podstawowe konstrukcje programistyczne ..... 208
14A. Podstawy pracy w środowisku C++ ..... 210
15A. Definiowanie funkcji obliczeniowych ..... 225

16A. Wyszukiwanie wzorca w tekście ..... 235
Infografika: Algorytm ..... 241
17A. Przetwarzanie napisów ..... 245
18A. Szyfrowanie i deszyfrowanie tekstu ..... 254

**Komunikacja i nauka w sieci** ..... 261
Infografika: ABC grywalizacji ..... 262
19. Kompetencje medialne i cyfrowe ..... 264
20. E-learning ..... 275

**Tworzenie stron internetowych** ..... 281
Infografika: Witryny i strony internetowe ..... 282
21. Projekt strony internetowej ..... 284
22. Podstawy pracy z HTML-em ..... 292
Infografika: Pozycjonowanie i optymalizacja stron WWW ..... 302
23. Podstawy pracy z CSS-em ..... 304
24. Elementy dynamiczne ..... 315
25. Publikacja i ocena strony WWW ..... 322

Indeks ..... 327

* W zależności od języka programowania preferowanego przez nauczyciela – do wyboru Python lub C++.

treści usunięte wg podstawy programowej 2024

# INFORMATYKA JEST WSZĘDZIE

**Informatyka zajmuje się przetwarzaniem informacji. Dzięki niej można rozwiązywać problemy za pomocą komputerów. Jest obecna niemal w każdej dziedzinie – wspomaga pracę pojedynczych osób i całych przedsiębiorstw, automatyzuje powtarzalne czynności, pozwala przeprowadzać symulacje skomplikowanych procesów.**




**Informatyka to nauka ścisła, która przydaje się osobom pracującym w różnych zawodach oraz w życiu codziennym.**

## O czym się uczymy na informatyce?



# JAK KORZYSTAĆ Z PODRĘCZNIKA

## Przemyśl problem i podaj rozwiązanie

**ZADANIE NA START**

**Poznasz książkę po okładce**

Wejdź na platformę **www.ted.com**, która gromadzi najciekawsze wykłady z cyklu TED Talks, i posłuchaj wykładu *Designing books is no laughing matter. OK, it is* (*Projektowanie książek to nic śmiesznego. No, może trochę*) wygłoszonego przez Chipa Kidda, słynnego amerykańskiego projektanta okładek. Z którymi tezami grafika się zgadzasz, a z którymi nie?



*samodzielna analiza problemu wprowadzająca w tematykę lekcji*

## Przeczytaj i dowiedz się więcej

**TWORZENIE KORESPONDENCJI SERYJNEJ**

Czasami trzeba się skomunikować z większą liczbą osób. Wówczas konieczne jest przygotowanie dużej liczby etykiet adresowych, listów, e-maili, raportów, zaświadczeń, kopert, zaproszeń i innych dokumentów różniących się tylko kilkoma szczegółami. Realizacja takiego zadania w przypadku tworzenia każdego pliku osobno wymagałaby ogromnych nakładów pracy i czasu – zwłaszcza gdyby się okazało, że na końcowym etapie trzeba wprowadzić jeszcze jakieś poprawki. Dlatego warto do tego celu wykorzystać korespondencję seryjną, dzięki której w kilka minut można wygenerować tyle dokumentów, ilu jest adresatów

*teoria*

**ZAPAMIĘTAJ!** *ważne informacje*

**W PRAKTYCE** *pomocne wskazówki*

**WIEM WIĘCEJ** *przydatne informacje*

**POSZUKAJ W INTERNECIE** *poszerzenie wiedzy*

## Wykonaj ćwiczenia

**Ćwiczenie 2.** **Projekt okładki**

Przygotuj w edytorze grafiki GIMP okładkę do książki i e-booka *Książę* Niccola Machiavellego, inną od prezentowanych powyżej. Tekst rozprawy znajdziesz w serwisie **https://wolnelektury.pl**.

1. Ustal format.
2. Zgromadź grafiki – własne lub gotowe. Pamiętaj o przepisach prawa autorskiego. Do portali zamieszczających grafikę na licencji Creative Commons CC należą m.in.
   - Wikimedia Commons – **https://commons.wikimedia.org**;
   - Openclipart Library – **https://openclipart.org**;
   - Pixabay – **https://pixabay.com**.

*zastosowanie omówionych zagadnień w ćwiczeniach wykonywanych krok po kroku*

## Uporządkuj swoją wiedzę i sprawdź umiejętności

**PODSUMOWANIE** *skrót najważniejszych wiadomości*

**PYTANIA SPRAWDZAJĄCE** *pytania weryfikujące stopień opanowania materiału*

**PYTANIA QUIZOWE** *krótkie pytania testowe*

**ZADANIA DODATKOWE** *zadania trudniejsze i rozszerzające temat*



# OCHRONA DANYCH I OPROGRAMOWANIA

Dynamiczny rozwój technologii często daje jej użytkownikom złudne poczucie bezpieczeństwa. Aby zapewnić ochronę zasobom przechowywanym na dysku, należy pamiętać o:

- odpowiedniej profilaktyce antywirusowej,
- bezpiecznych hasłach,
- aktualizowaniu oprogramowania,
- tworzeniu kopii zapasowych.

# OCHRONA DANYCH OSOBOWYCH W SIECI

## RODZAJE DANYCH OSOBOWYCH

**DANE OSOBOWE**

**ZWYKŁE**

- imię i nazwisko
- PESEL
- dane o lokalizacji
- identyfikator internetowy, np. adres IP

**WRAŻLIWE**

- pochodzenie rasowe lub etniczne
- poglądy polityczne
- przekonania religijne lub światopoglądowe
- przynależność do partii, związków zawodowych
- dane genetyczne i biometryczne
- dane genetyczne i biometryczne

## PSEUDONIMIZACJA I ANONIMIZACJA

**Marcin Nowak,**

urodzony

12 kwietnia 2004

w Warszawie,

ukończył kurs na

patent sternika

motorowodnego

Pseudonimizacja to takie przetwarzanie danych osobowych, aby niemożliwe było zidentyfikowanie osoby na podstawie tych danych bez dodatkowego klucza, który musi być przechowywany w innym miejscu. Jest to proces odwracalny.

MN,

2004,

PSM

Anonimizacja to usunięcie wszystkich informacji umożliwiających w jakikolwiek sposób identyfikację osoby, której dane dotyczą. Jest to proces nieodwracalny.

[redacted]

urodzony

12 kwietnia 2004

w Warszawie,

ukończył kurs na

patent sternika

motorowodnego

**Dane osobowe to informacje, które pozwalają jednoznacznie zidentyfikować osobę. Wykorzystywanie (przetwarzanie) danych osobowych uregulowano w Rozporządzeniu o Ochronie Danych Osobowych, powszechnie znanym jako RODO, które obowiązuje w całej Unii Europejskiej od 25 maja 2018 roku.**

## PAKIET PRAW KONSUMENTA

Wraz z wprowadzeniem RODO każdy użytkownik internetu otrzymał **prawa**, dzięki którym może zwiększyć ochronę swoich danych osobowych w sieci.



prawo dostępu do informacji na temat rodzaju danych i celu ich przetwarzania



prawo do poprawiania danych



prawo do bycia zapomnianym, czyli do usunięcia danych z bazy



prawo do ograniczenia przetwarzania danych



prawo do przenoszenia danych





### PRACA Z INFOGRAFIKĄ

1. W jakim celu wprowadzono RODO?
2. Jakie dane osobowe są niezmienne, a jakie się zmieniają? Podaj kilka przykładów.




# 1. Bezpieczna praca z komputerem

- Dobre praktyki w zakresie ochrony oprogramowania
- Sposoby zabezpieczania kont
- Kopie zapasowe, aktualizacje, szyfrowanie

## ZADANIE NA START

### Jak szybko komputer złamie hasło

Aby sprawdzić, ile maksymalnie prób potrzeba, aby złamać czterocyfrowy PIN do karty kredytowej, należy wyznaczyć liczbę wszystkich możliwych kodów w zależności od tego, czy w kodzie:

- cyfry mogą się powtarzać ($10 \cdot 10 \cdot 10 \cdot 10 = 10\,000$);
- cyfry mogą się powtarzać, ale nie czterokrotnie ($10 \cdot 10 \cdot 10 \cdot 10 - 10 = 9990$);
- cyfry nie mogą się powtarzać ($10 \cdot 9 \cdot 8 \cdot 7 = 5040$).

Oblicz, ile maksymalnie prób potrzeba, aby złamać czteroznakowy kod złożony z małych i wielkich liter alfabetu łacińskiego.




## BEZPIECZNIE W PRACOWNI KOMPUTEROWEJ

W sali informatycznej obowiązuje regulamin uwzględniający specyfikę pracy z komputerem w szkole. Pamiętaj, że należy:

- korzystać z urządzeń zgodnie ze wskazówkami nauczyciela;
- informować nauczyciela o wszelkich usterkach zauważonych podczas pracy;
- logować się tylko na własne konto użytkownika;
- przechowywać dokumenty w miejscu wskazanym przez nauczyciela;
- dbać o porządek na stanowisku pracy.

## OCHRONA SYSTEMU I DANYCH ZAPISANYCH NA KOMPUTERZE

Bezpieczeństwo komputerowe to niezwykle istotna sprawa. Komputer i dane na nim zgromadzone są narażone na działanie złośliwego oprogramowania (*malware*) oraz na innego rodzaju ataki cyberprzestępców.

**ZAPAMIĘTAJ!**

**Złośliwe oprogramowanie** obejmuje różne typy programów, których celem jest działanie na szkodę użytkowników, np. poprzez kradzież danych czy też przejęcie kontroli nad komputerem.

Do infekcji może dojść m.in. przez:

- użycie zainfekowanego nośnika;
- odwiedzenie witryny zawierającej kod, który z powodu błędu w zabezpieczeniach danej przeglądarki instaluje złośliwe oprogramowanie bez wiedzy użytkownika;
- pobranie zainfekowanego pliku, np. programu czy filmu;
- otworzenie załącznika zawierającego złośliwy kod;
- kliknięcie w portalu społecznościowym zdjęcia z ukrytym rozszerzeniem, które uruchamia złośliwy program.

### Ćwiczenie 1. Rodzaje złośliwego oprogramowania

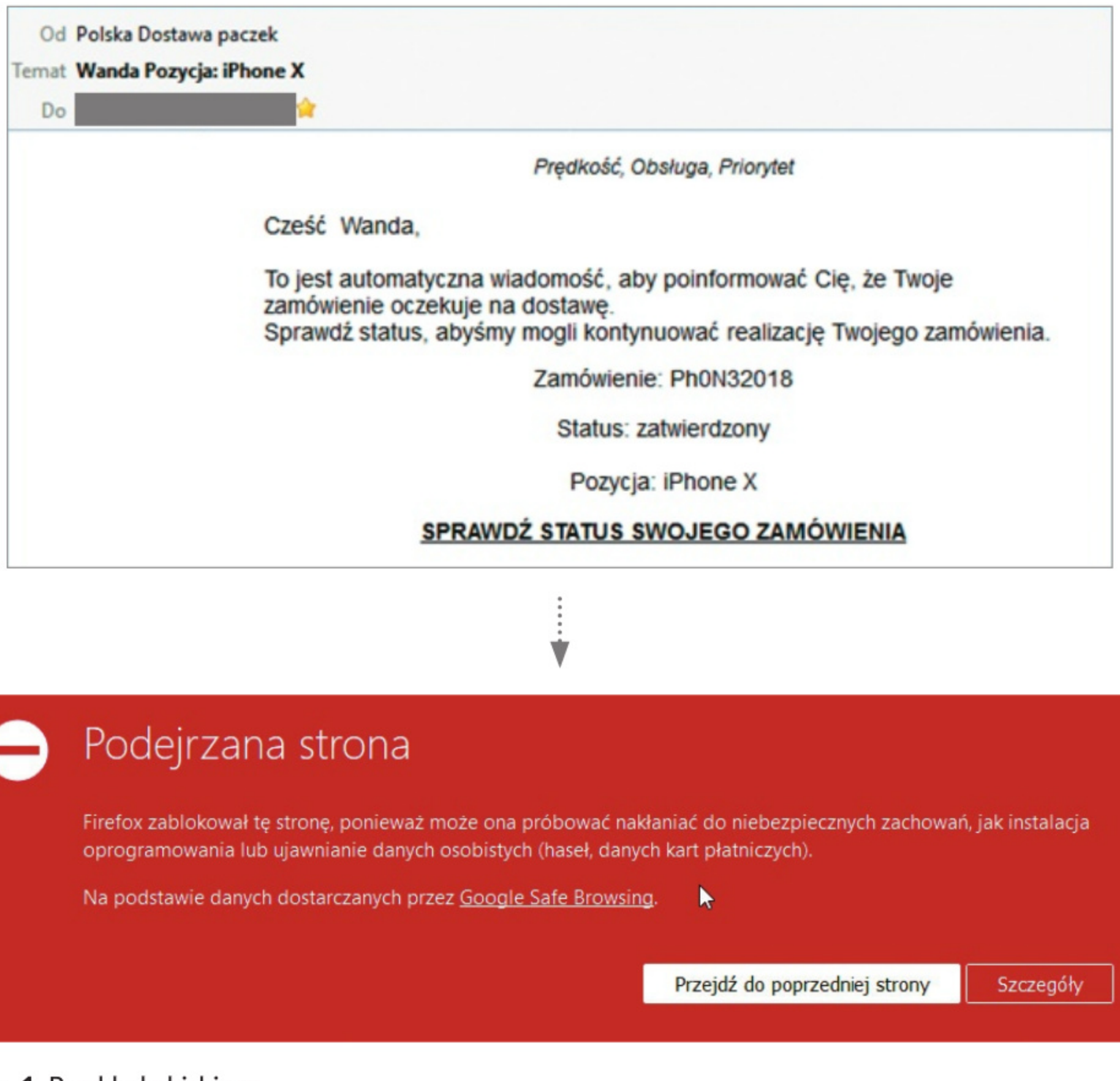
Zapoznaj się z poniższą grafiką, a następnie scharakteryzuj krótko wymienione grupy złośliwego oprogramowania. Potrzebne informacje wyszukaj w internecie.



Cyberprzestępcy nie zawsze sięgają po specjalne programy. Niektórzy wykorzystują luki systemowe lub po prostu do skutku odgadują hasło dostępu (tzw. ataki *brute force*). Inni wyłudzają poufne dane przez zastosowanie phishingu. Ta metoda wykorzystuje socjotechnikę. Najczęstszą jej formą są e-maile udające wiadomości od danego nadawcy i kierujące na odpowiednio spreparowane strony internetowe, które mają nakłonić niedoświadczonego użytkownika do ujawnienia informacji otwierających dostęp do konta bankowego, PayPala czy karty kredytowej. Wiele firm publikuje w swoich serwisach informacje na temat oszustw, w ramach których wykorzystywane są ich nazwa i logo (np. Google, DPD, OLX).

**POSZUKAJ W INTERNECIE**

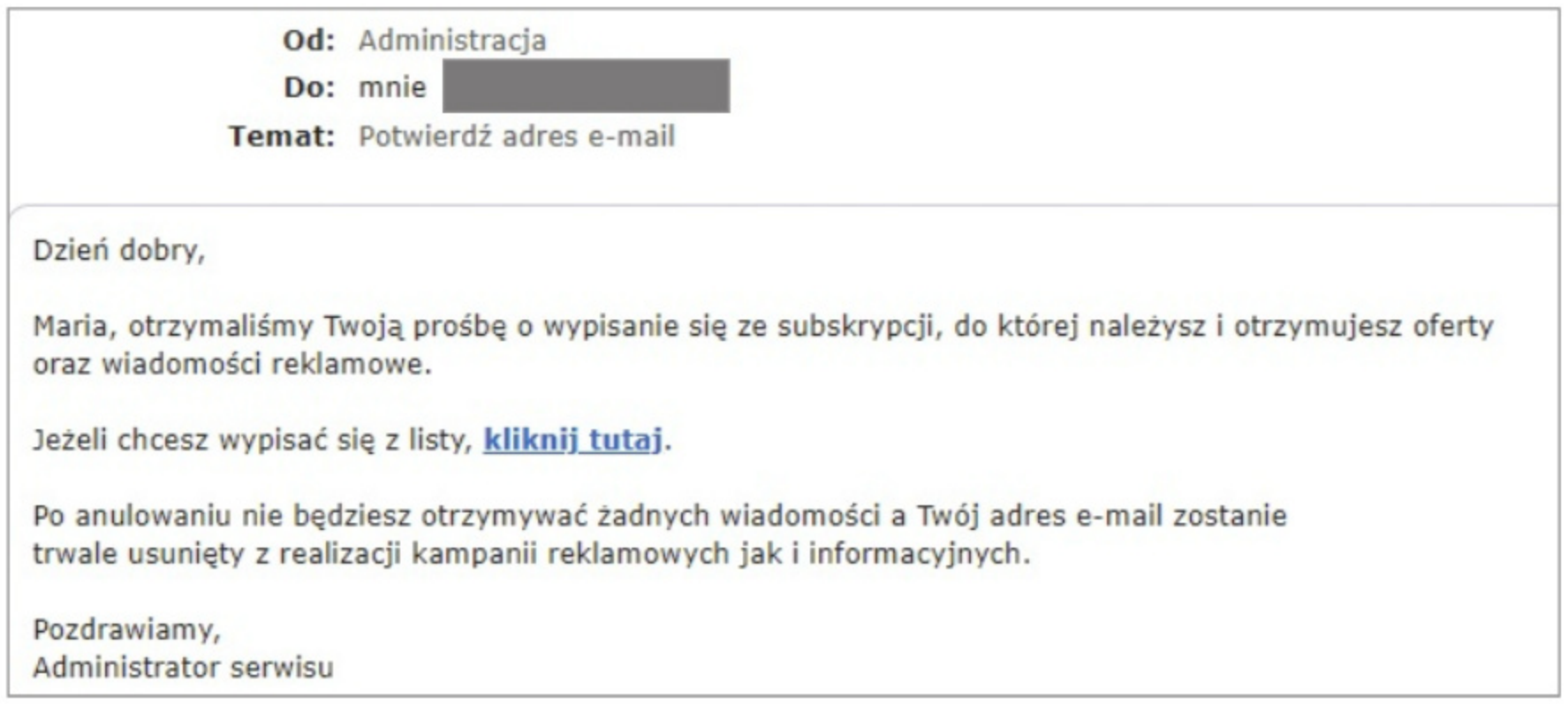
Sprawdź, na czym polegają oszustwa komputerowe typu spoofing i pharming.




**Rys. 1.** Przykład phishingu

## Ćwiczenie 2. Phishing

Scharakteryzuj język zamieszczonych na str. 13 listów i ich nadawców. Poszukaj w **Bazie wiedzy** w serwisie **gov.pl** informacji, jak rozpoznać e-mail wyłudzający dane, i określ, co sprawia, że te wiadomości noszą znamiona phishingu.



**Od:** Administracja
**Do:** mnie [redacted]
**Temat:** Potwierdź adres e-mail

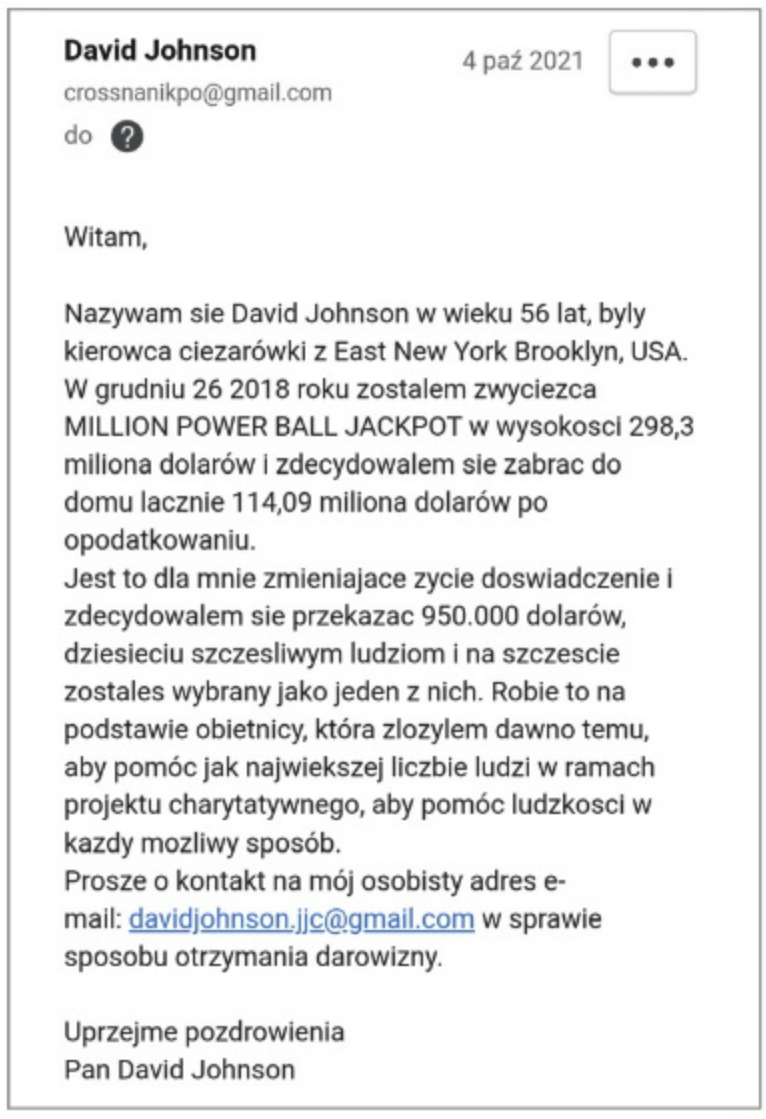
Dzień dobry,

Maria, otrzymaliśmy Twoją prośbę o wypisanie się ze subskrypcji, do której należysz i otrzymujesz oferty oraz wiadomości reklamowe.

Jeżeli chcesz wypisać się z listy, **kliknij tutaj**.

Po anulowaniu nie będziesz otrzymywać żadnych wiadomości a Twój adres e-mail zostanie trwale usunięty z realizacji kampanii reklamowych jak i informacyjnych.

Pozdrawiamy,
Administrator serwisu



**David Johnson** 4 paź 2021
crossnanikpo@gmail.com
do ?

Witam,

Nazywam sie David Johnson w wieku 56 lat, byly kierowca ciezarówki z East New York Brooklyn, USA. W grudniu 26 2018 roku zostalem zwyciezca MILLION POWER BALL JACKPOT w wysokosci 298,3 miliona dolarów i zdecydowalem sie zabrac do domu lacznie 114,09 miliona dolarów po opodatkowaniu.
Jest to dla mnie zmieniajace zycie doswiadczenie i zdecydowalem sie przekazac 950.000 dolarów, dziesieciu szczesliwym ludziom i na szczescie zostales wybrany jako jeden z nich. Robie to na podstawie obietnicy, która zlozylem dawno temu, aby pomóc jak najwiekszej liczbie ludzi w ramach projektu charytatywnego, aby pomóc ludzkosci w kazdy mozliwy sposób.
Prosze o kontakt na mój osobisty adres e-mail: davidjohnson.jjc@gmail.com w sprawie sposobu otrzymania darowizny.

Uprzejme pozdrowienia
Pan David Johnson



**Od:** Transakcja otwarta48
**Do:** mnie [redacted]
**Temat:** Masz do czynienia z 97 nierozliczoną transakcją [redacted] [DSsFGUlb2T]

To jest automatyczna wiadomość dotycząca Twojego salda w the Bitcoin Code.

bitcoin

**Twoja wypłata jest gotowa!**

Sz.P. właściciel konta,

Dziękujemy za zainteresowanie naszym programem Bitcoin.
Pragniemy poinformować, iż Państwa bonus Bitcoin czeka już na wypłatę.

**Informacja o koncie #4589mBTC**

Imię: *Prywatny
E-mail: *Prywatny
Kraj: Polska
Saldo w Bitcoinach: 77086,70 zł

Potwierdź teraz wypłatę

Aby zapewnić systemowi oraz danym właściwą ochronę, należy:

- systematycznie przeprowadzać aktualizacje – wypełniać luki w zabezpieczeniach;
- aktywować systemową zaporę sieciową lub zainstalować preferowaną zaporę – zezwolić na monitorowanie pakietów danych przesyłanych między komputerem a siecią i na blokowanie nieautoryzowanego dostępu do komputera;
- zainstalować oprogramowanie typu *internet security* – podstawową ochronę zapewnia skaner antywirusowy, który sprawdza pliki zapisane na komputerze i w razie potrzeby usuwa zagrożenie; takie oprogramowanie ma także często funkcje zaawansowane dostępne w zależności od wybranego pakietu, np. moduł ochrony poczty elektronicznej, filtr antyspamowy, blokowanie niebezpiecznych stron, ochronę płatności dokonywanych online, skaner linków z portali społecznościowych, szyfrowanie danych, lokalizowanie skradzionego komputera;
- skanować programem antywirusowym nośniki zewnętrzne;
- nie odwiedzać witryn kwalifikowanych przez przeglądarkę jako niebezpieczne;

- nie pobierać plików z niesprawdzonych serwisów;
- nie otwierać wiadomości z nieznanych źródeł, o podejrzanie brzmiących tematach, z błędami ortograficznymi bądź językowymi;
- nie klikać linków w wiadomościach od nieznanych nadawców.

## BEZPIECZNE HASŁO

Hasła potrzebne są wszędzie – aby dostać się do poczty elektronicznej na serwerze, do portalu społecznościowego albo konta bankowego, sprawdzić historię zakupów w sklepie internetowym, zarezerwować z domu książkę w bibliotece czy pracować wspólnie z kolegami w chmurze.

Jak pokazują badania, przeciętny użytkownik internetu korzysta z dwudziestu kilku haseł do różnego rodzaju aktywności. Trudno je wszystkie zapamiętać, zwłaszcza te rzadko używane. Zapomniane hasło można zmienić z wykorzystaniem specjalnego formularza na stronie. Wysyłany jest wtedy e-mail w celu weryfikacji użytkownika. Dobrym rozwiązaniem okazuje się korzystanie z menedżera haseł, który przechowuje wszystkie hasła w jednym miejscu. Wystarczy pamiętać jedno hasło, które daje dostęp do aplikacji. Ponadto zwykle menedżery haseł oferują też generowanie losowych haseł.

Bezpieczne hasło jest łatwe do zapamiętania dla użytkownika, lecz trudne do odgadnięcia przez inne osoby. Powinno się składać z minimum ośmiu znaków (długie hasło sprawia, że liczba możliwych kombinacji wzrasta i wydłuża się czas potrzebny hakerowi na jego złamanie), zawierać małe i wielkie litery, cyfry oraz znaki specjalne. Jeśli korzystasz z serwisu bez podawania ważnych danych, możesz zastosować prostsze hasło.

### Ćwiczenie 3. Samodzielne tworzenie silnego hasła

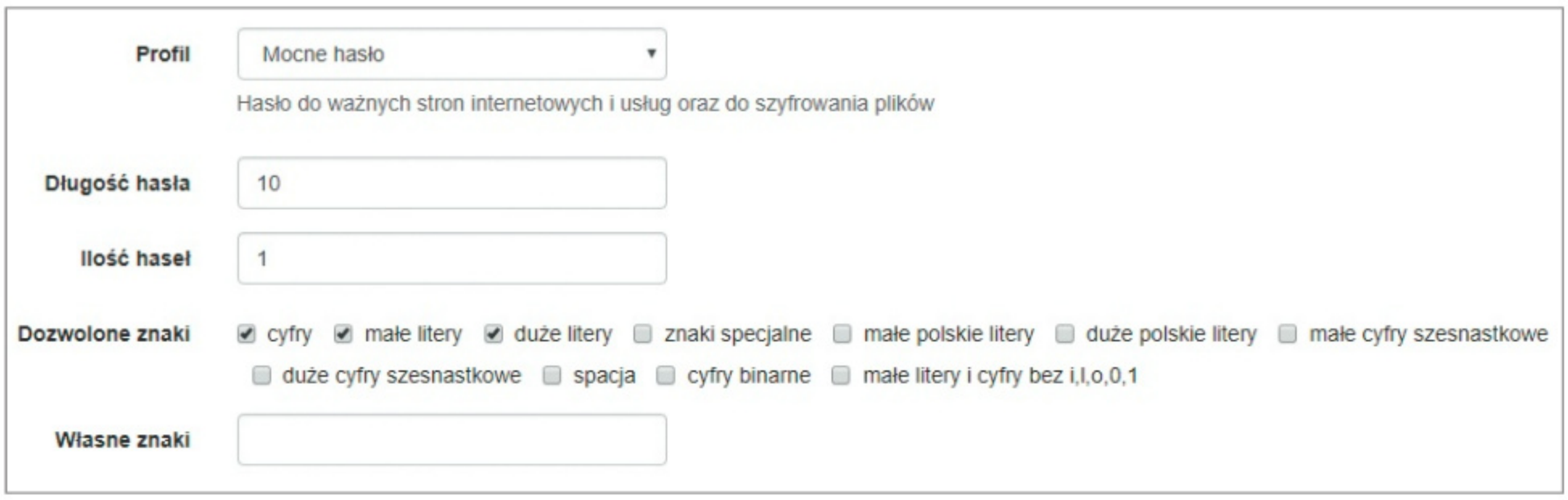
Utwórz silne hasło, którego podstawę będzie stanowić fraza łatwa do zapamiętania.

1. Wybierz podstawę hasła, np. przysłowie: „Idzie luty, podkuj buty”.
2. Określ zasady tworzenia hasła, czyli sposób, który pozwoli ci je odtworzyć. Zasady mogą być np. takie:
   - pomiń wyrazy jednoliterowe;
   - z każdego wyrazu weź dwie pierwsze litery;
   - jeśli któraś z liter nie należy do alfabetu łacińskiego (np. ą, ó), zamień ją na odpowiadającą jej literę alfabetu łacińskiego;
   - każdą parę liter zapisz w formacie wielka–mała (np. Wi);
   - po pierwszych dwóch literach i na końcu hasła wstaw znaki specjalne (np. #, *);
   - dwie ostatnie litery poprzedź cyfrą.
3. Zastosuj przyjęte zasady do wybranej podstawy hasła. Dla przysłowia wcześniej wskazanego hasło to np. Id#LuPo7Bu*.

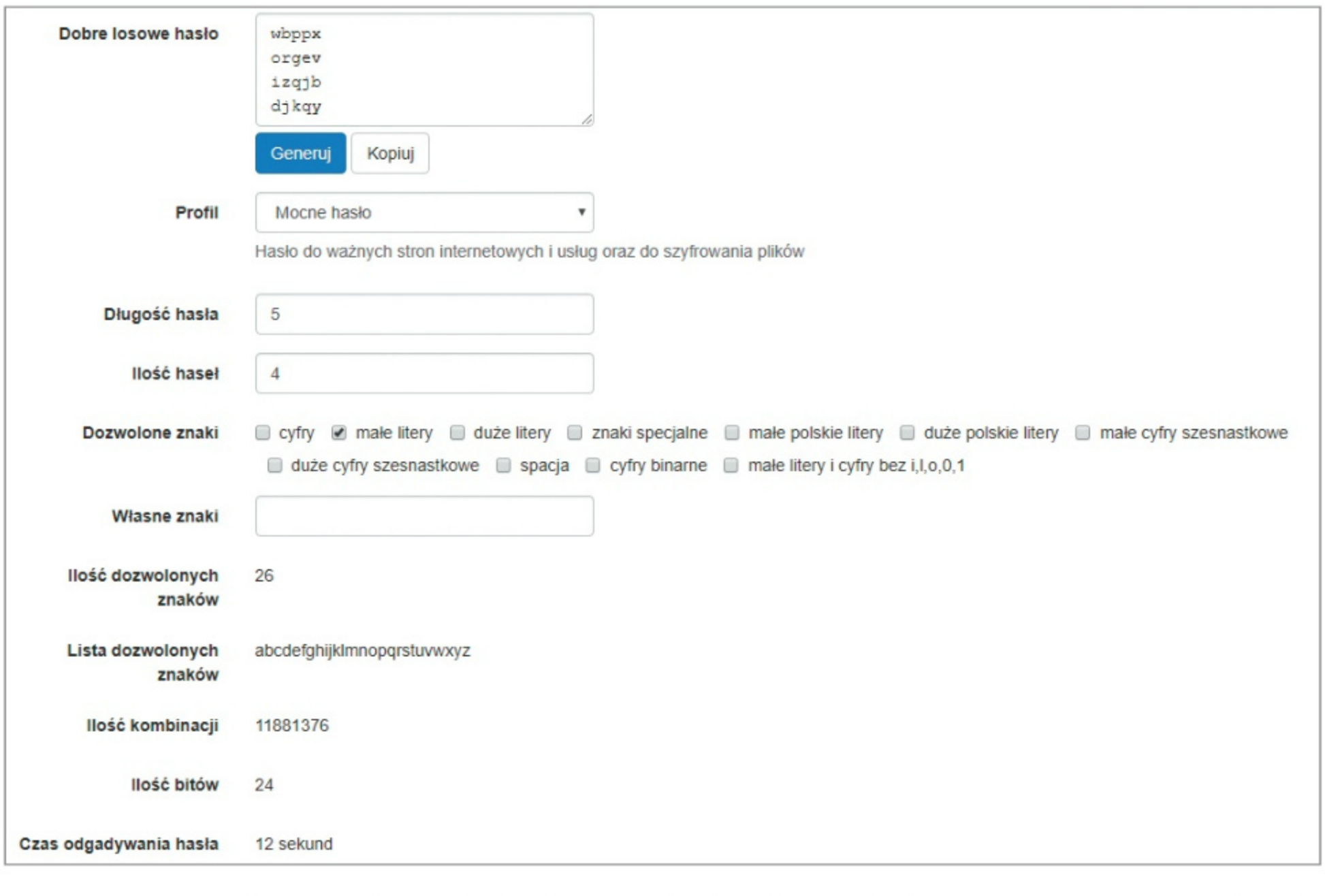
## Ćwiczenie 4. Tworzenie hasła za pomocą generatora

Skorzystaj z generatora haseł na stronie **http://dobrehaslo.pl** i utwórz mocne hasło, z uwzględnieniem różnych parametrów, a następnie przeanalizuj otrzymane wyniki.

1. Wybierz profil hasła i określ parametry – długość, znaki dozwolone, znaki własne.




2. Dokonaj analizy otrzymanych wyników, np. takiej:
   - w przypadku hasła złożonego z pięciu małych liter komputer musi sprawdzić 11 881 376 kombinacji, co zajmie mu zaledwie 12 s;




   - w przypadku zwiększenia liczby znaków hasła do siedmiu oraz liczby znaków do wyboru do 93 (cyfry, małe i wielkie litery, znaki specjalne) komputer musi sprawdzić już ponad 60 bilionów kombinacji. Warto zdecydować się na co najmniej osiem znaków, ponieważ odgadywanie takiego hasła wydłuży się znacząco.

**WIEM WIĘCEJ**

Coraz częściej oprócz albo zamiast haseł stosuje się **zabezpieczenia biometryczne**, tj. oparte na danych biometrycznych użytkownika, takich jak zapis geometrii twarzy lub dłoni, linii papilarnych albo obrazu tęczówki. W ten sposób można ograniczyć dostęp do szczególnie wrażliwych danych lub pomieszczeń – np. Muzeum Śląskie w Katowicach stosuje uwierzytelnianie za pomocą czytników wymagających wprowadzenia PIN-u oraz zeskanowania karty i linii papilarnych. Smartfony i tablety umożliwiają odblokowanie ekranu wzorem twarzy – wystarczy patrzeć na ekran urządzenia.

## Ćwiczenie 5. Testowanie haseł

Na podstawie wymagań opisanych na stronie **www.passwordmeter.com** dokonaj analizy podanych haseł: Br4aKtm@, kotek7, nap@rawa4, Duch09@21.

1. Wpisz hasło w pole **Password** (warto zaznaczyć opcję **Hide**, która maskuje tekst), aby program zaprezentował jego diagnozę w postaci punktacji procentowej (**Score**) oraz słownej (**Complexity**; hasło bardzo słabe – *very weak*, słabe – *weak*, dobre – *good*, silne – *strong*, bardzo silne – *very strong*).

| Test Your Password | | Minimum Requirements |
|---|---|---|
| **Password:** | Br4aKtm@ | • Minimum 8 characters in length<br>• Contains 3/4 of the following items:<br>- Uppercase Letters<br>- Lowercase Letters<br>- Numbers<br>- Symbols |
| **Hide:** | ☐ | |
| **Score:** | 72% | |
| **Complexity:** | Strong | |

2. Punkty są dodawane za spełnienie minimalnych wymagań, liczbę znaków, występowanie wielkich i małych liter, cyfr oraz znaków specjalnych, umieszczenie cyfr lub znaków specjalnych w środku hasła.

| Additions | | Type | Rate | Count | Bonus |
|---|---|---|---|---|---|
| ✓ | Number of Characters | Flat | +(n*4) | 8 | + 32 |
| ★ | Uppercase Letters | Cond/Incr | +((len-n)*2) | 2 | + 12 |
| ★ | Lowercase Letters | Cond/Incr | +((len-n)*2) | 4 | + 8 |
| ✓ | Numbers | Cond | +(n*4) | 1 | + 4 |
| ✓ | Symbols | Flat | +(n*6) | 1 | + 6 |
| ✓ | Middle Numbers or Symbols | Flat | +(n*2) | 1 | + 2 |
| ★ | Requirements | Flat | +(n*2) | 5 | + 10 |

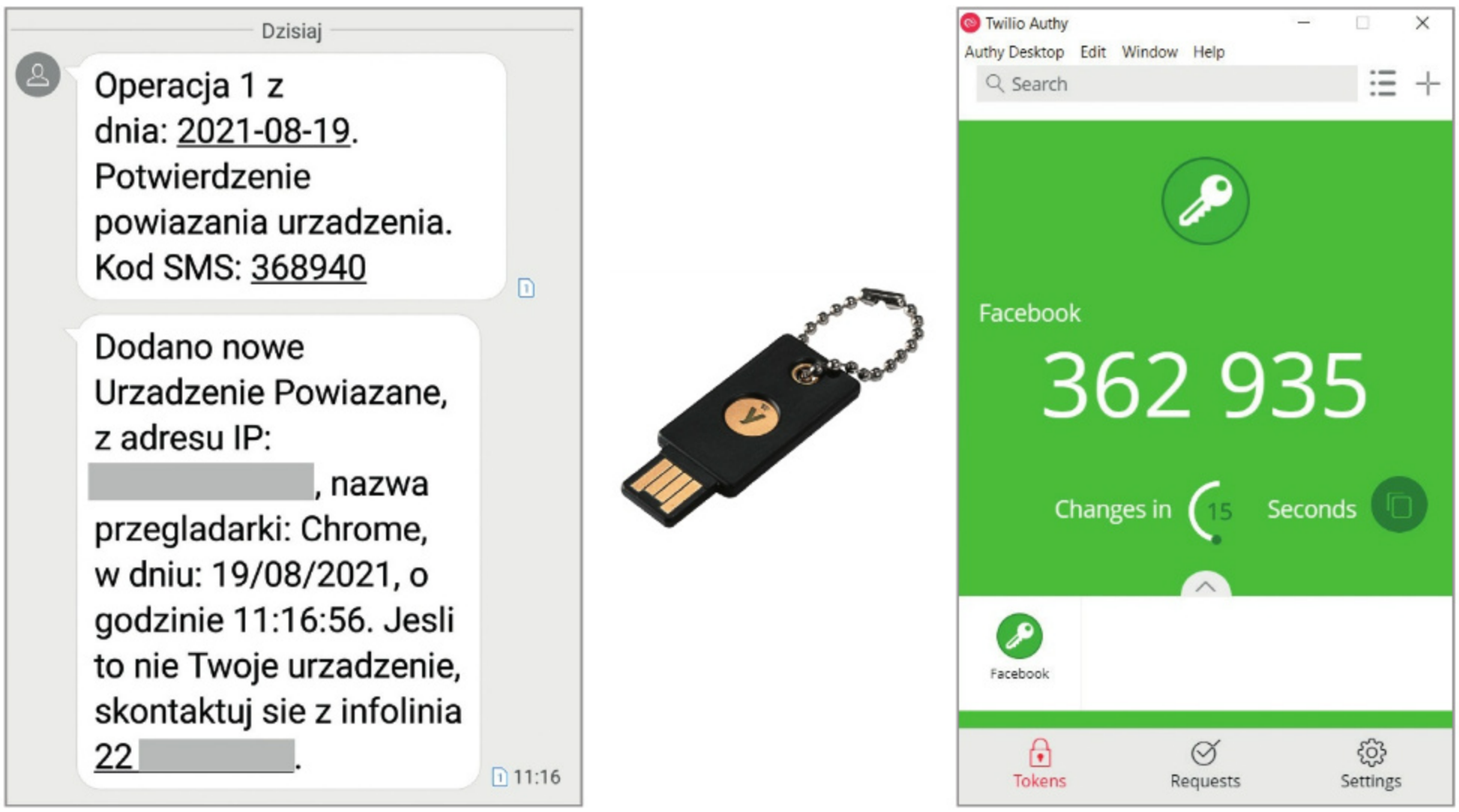
3. Punkty odejmowane są za użycie samych liter lub cyfr, powtarzające się litery, cyfry lub znaki specjalne (np. **a**Rw4*__a__) albo następujące po sobie dowolne małe litery (jak

na poniższym zrzucie), wielkie litery lub cyfry, co najmniej trzy kolejne litery, cyfry lub znaki specjalne (np. abc, 345).

| Deductions | | | | | |
|---|---|---|---|---|---|
| ✓ | Letters Only | Flat | -n | 0 | 0 |
| ✓ | Numbers Only | Flat | -n | 0 | 0 |
| ✓ | Repeat Characters (Case Insensitive) | Comp | - | 0 | 0 |
| ✓ | Consecutive Uppercase Letters | Flat | -(n*2) | 0 | 0 |
| ! | Consecutive Lowercase Letters | Flat | -(n*2) | 1 | - 2 |
| ✓ | Consecutive Numbers | Flat | -(n*2) | 0 | 0 |
| ✓ | Sequential Letters (3+) | Flat | -(n*3) | 0 | 0 |
| ✓ | Sequential Numbers (3+) | Flat | -(n*3) | 0 | 0 |
| ✓ | Sequential Symbols (3+) | Flat | -(n*3) | 0 | 0 |

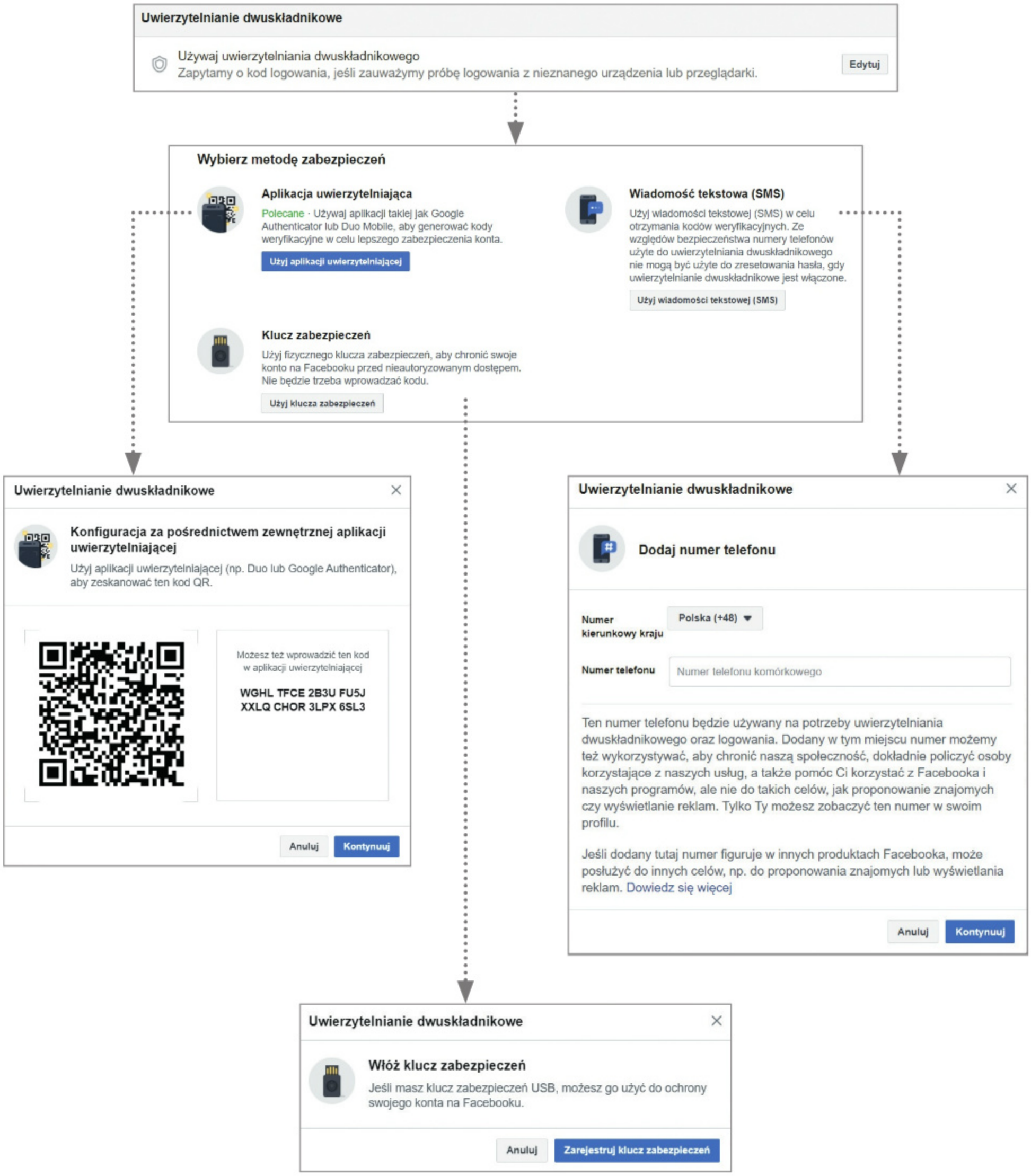
## UWIERZYTELNIANIE DWUSKŁADNIKOWE

Uwierzytelnianie dwuskładnikowe wzmacnia ochronę konta. Podczas procesu logowania trzeba standardowo podać login i hasło, a następnie dodatkowo potwierdzić swoją tożsamość innym kanałem, np. za pomocą kodu otrzymanego SMS-em lub wygenerowanego z zewnętrznej aplikacji uwierzytelniającej (desktopowej lub mobilnej) albo przy użycia klucza zabezpieczeń.




**Rys. 2.** Trzy metody dodatkowej warstwy zabezpieczenia logowania: kod przesłany SMS-em, klucz zabezpieczeń, kod wygenerowany w aplikacji

Aby korzystać z uwierzytelniania dwuskładnikowego na Facebooku, w ustawieniach konta należy rozwinąć menu **Konto**, a następnie wybrać kolejno: **Ustawienia i prywatność → Ustawienia → Bezpieczeństwo i logowanie → Uwierzytelnianie dwuskładnikowe → Używaj uwierzytelniania dwuskładnikowego**. Po kliknięciu polecenia **Edytuj** można zaznaczyć najwygodniejszą opcję.




**Rys. 3.** Ustawianie uwierzytelniania dwuskładnikowego na Facebooku

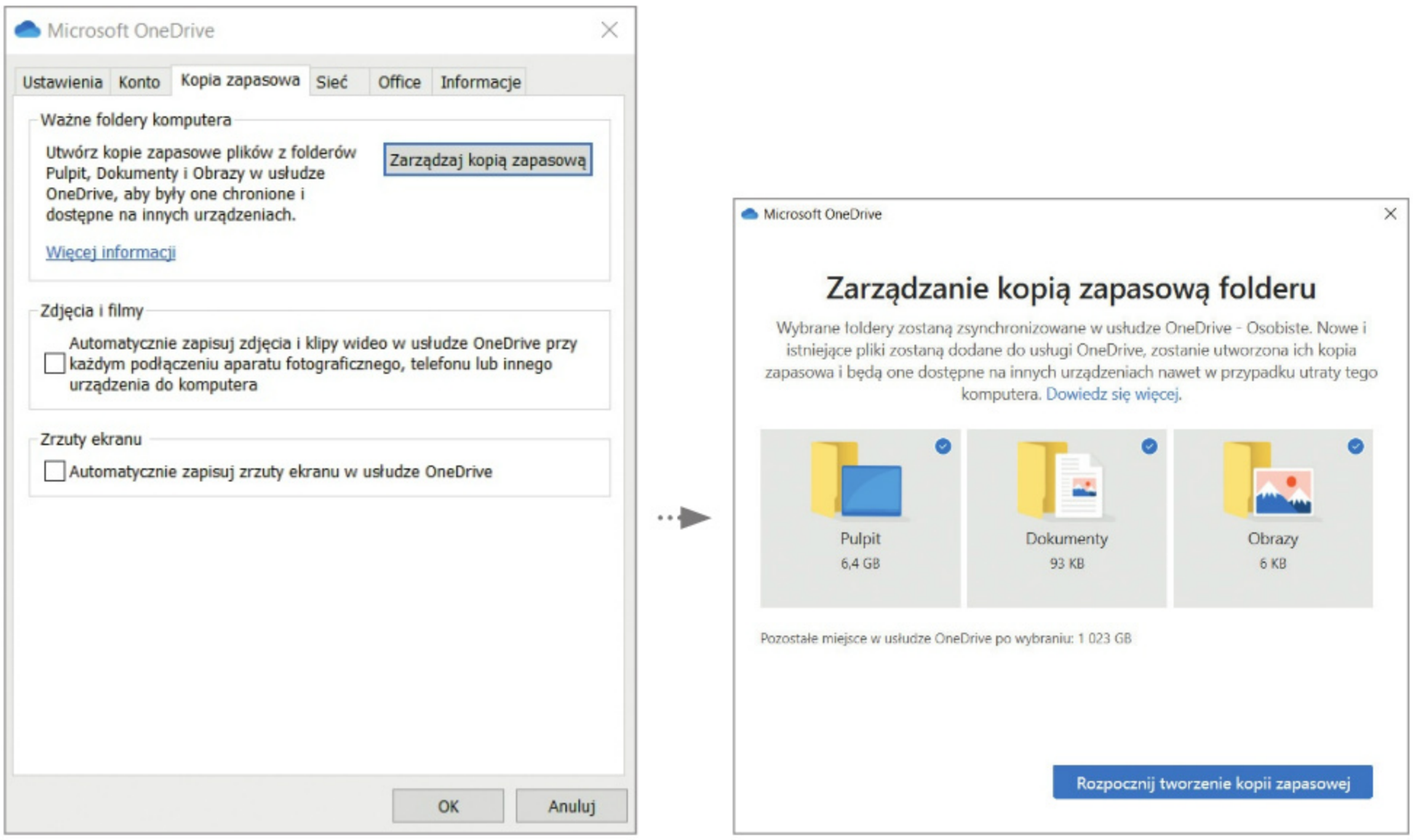
## Ćwiczenie 6. Uwierzytelnianie dwuskładnikowe

Jakie dostrzegasz zalety i wady uwierzytelniania dwuskładnikowego?

## JAK MOŻNA ZWIĘKSZYĆ BEZPIECZEŃSTWO?

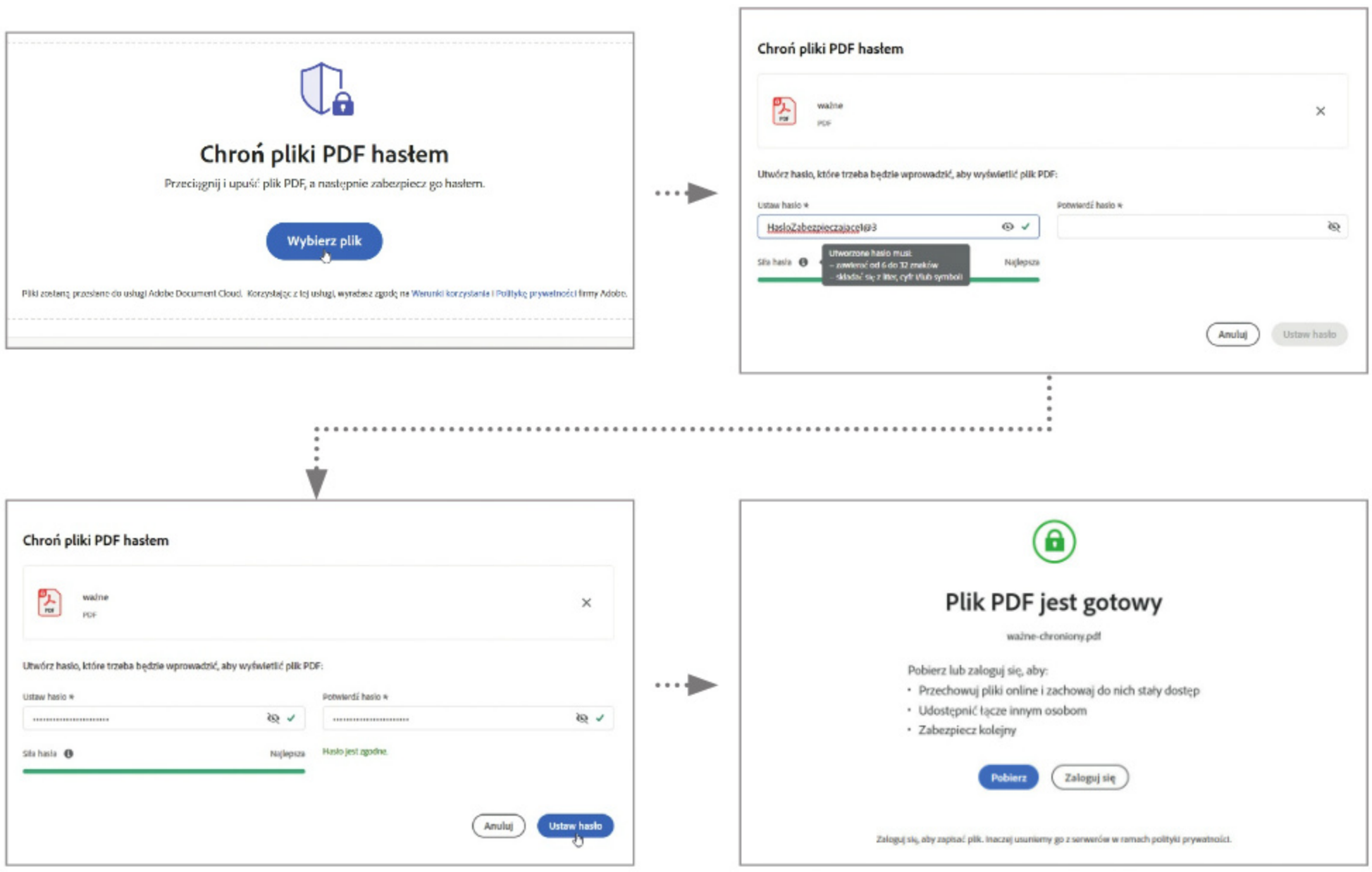
Po pierwsze, należy aktualizować oprogramowanie – zarówno systemowe, jak i użytkowe, np. przeglądarkę stron WWW. Aktualizacja zawiera poprawki (tzw. łatki), które usprawniają działanie programów poprzez dodawanie nowych funkcji oraz eliminowanie usterek zgłoszonych w ramach jawnego i niejawnego testowania błędów lub przez przypadkowych użytkowników. W ten sposób są usuwane luki bezpieczeństwa, które mogłyby zostać wykorzystane przez cyberprzestępców.

Po drugie, trzeba tworzyć kopie zapasowe zawierające informacje i pliki niezbędne do uruchomienia systemu oraz do odtworzenia oryginalnych danych w razie ich utraty bądź uszkodzenia. Regularnie zapisywane kopie bezpieczeństwa wybranych folderów lub dysków albo całego systemu (obraz systemu) pozwalają uniknąć nieodwracalnej utraty danych i zminimalizować konieczność korzystania z usług serwisów komputerowych. Kopie można zapisywać i przechowywać na różnych nośnikach, np. na dyskach zewnętrznych albo w chmurze.




**Rys. 4.** Tworzenie kopii zapasowej na dysku OneDrive

Niekiedy warto również rozważyć szyfrowanie dokumentów, folderów bądź dysków. Szyfrowanie polega na przekształcaniu wiadomości, tzw. tekstu jawnego, w szyfrogram tak, by uniemożliwić osobom niepowołanym odczytanie informacji. W przypadku prostego szyfrowania hasłem osoby znające to hasło będą mogły uzyskać dostęp do danego pliku lub folderu. Zaawansowane szyfrowanie oznacza, że zaszyfrowane dane są chronione kluczem, a ich odblokowanie następuje dopiero po wprowadzeniu właściwego hasła lub kodu.




**Rys. 5.** Proste szyfrowanie dokumentu PDF hasłem za pomocą bezpłatnej usługi online Adobe Acrobat

Na koniec warto zadbać o odpowiednie ustawienia przeglądarki, świadomie wyrażać zgodę na śledzenie lokalizacji oraz korzystanie z plików cookies (tzw. ciasteczka zwykle służą do gromadzenia danych statystycznych pozwalających identyfikować i optymalizować sposób korzystania ze stron internetowych), a także rozważyć zainstalowanie oprogramowania blokującego reklamy i złośliwe oprogramowanie. Poza tym osoby, które korzystają ze współdzielonego komputera czy tabletu, mogą pracować w trybie domyślnym i po zakończeniu przeglądania wyczyścić historię albo od razu uruchomić przeglądarkę w trybie prywatnym.

Trzeba pamiętać, że tryb prywatny nie oznacza anonimowości. Aktywność użytkownika zawsze jest widoczna dla administratora sieci i dostawcy usług internetowych. Każdy użytkownik internetu jest odpowiedzialny za to, co robi w sieci.

## PODSUMOWANIE

- Dane osobowe, np. imię i nazwisko, PESEL, pozwalają jednoznacznie zidentyfikować określoną osobę. Serwisy WWW nie mogą gromadzić i udostępniać danych osobowych użytkowników bez ich wiedzy i zgody. Zgodę na przetwarzanie danych w każdej chwili można wycofać.
- Komputer i dane narażone są na działanie złośliwego oprogramowania (m.in. wirusów, robaków, trojanów) oraz innego typu ataki hakerów (m.in. phishing). Aby zapewnić im należytą ochronę, należy zainstalować zaporę sieciową

i oprogramowanie ochronne, w tym antywirusowe, oraz systematycznie aktualizować system operacyjny.

- Bezpieczne hasło powinno się składać z minimum ośmiu znaków, zawierać małe i wielkie litery, cyfry oraz znaki specjalne.
- Uwierzytelnianie dwuskładnikowe to funkcja zabezpieczeń, która wzmacnia ochronę konta. Polega na dodatkowym potwierdzeniu tożsamości innym kanałem, np. za pomocą kodu otrzymanego SMS-em lub wygenerowanego z aplikacji uwierzytelniającej albo przy użyciu klucza zabezpieczeń.
- Aby zwiększyć bezpieczeństwo systemu operacyjnego oraz przechowywanych danych, należy aktualizować oprogramowanie i tworzyć kopie zapasowe.

## PYTANIA SPRAWDZAJĄCE

1. Czy PESEL należy do danych osobowych? Uzasadnij.
2. Jakie konsekwencje może mieć brak stosowania środków ochrony komputera oraz danych? Podaj cztery przykłady.
3. Jakie zasady trzeba uwzględnić przy ustalaniu hasła, aby stało się ono trudne do złamania?

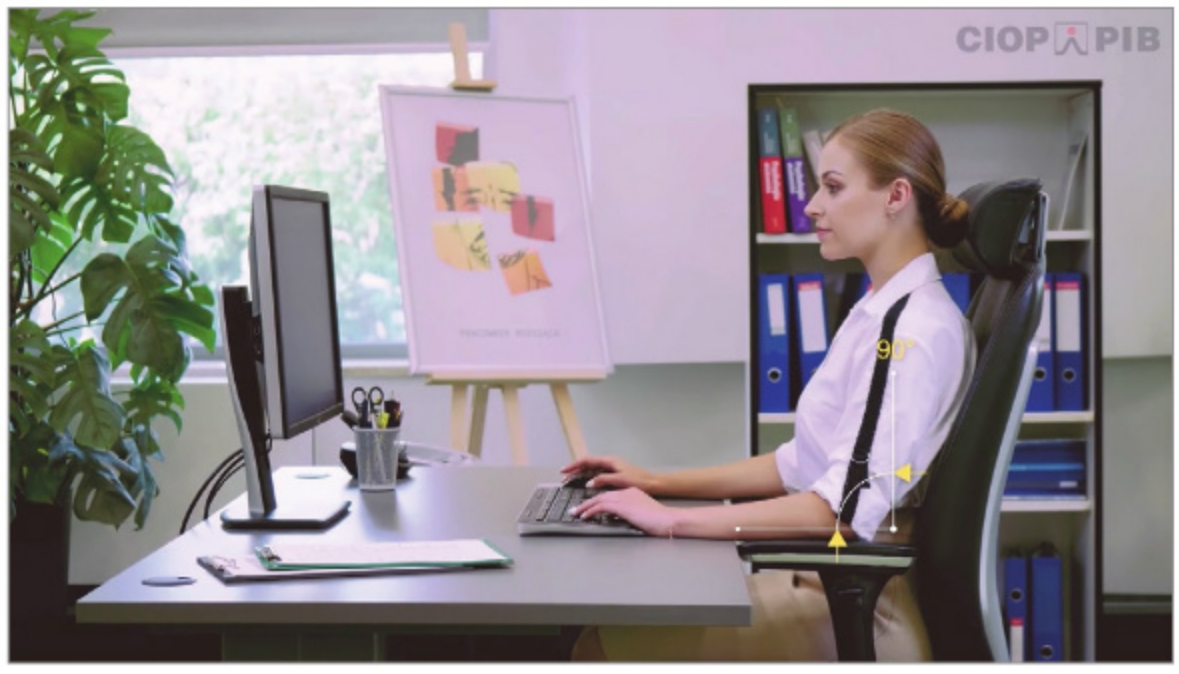
## PYTANIA QUIZOWE

1. Na czym polega pseudonimizacja danych?
   **a.** Na usunięciu wszystkich informacji umożliwiających identyfikację osoby.
   **b.** Na usunięciu informacji pozwalających na identyfikację osoby bez dodatkowych danych.
   **c.** Na zastąpieniu nazwiska pierwszą literą.
   **d.** Na dodaniu dodatkowych informacji z innego źródła.
2. Czym charakteryzuje się bezpieczne hasło?
   **a.** Jest krótkie – nikt nie podejrzy, gdy je wpisujemy.
   **b.** Jest złożone z samych cyfr.
   **c.** Ma nie więcej niż 4 znaki.
   **d.** Jest trudne do odgadnięcia przez postronne osoby.
3. Jak przebiega proces logowania z uwierzytelnianiem dwuskładnikowym?
   **a.** Podaje się login, hasło i drugie hasło, mocniejsze od pierwszego.
   **b.** Podaje się login i hasło, a następnie dzwoni do administratora serwisu z prośbą o dostęp.
   **c.** Podaje się login i hasło, a następnie potwierdza swoją tożsamość innym kanałem, np. za pomocą kodu otrzymanego SMS-em.
   **d.** Podaje się login i hasło, a następnie skanuje kod QR na stronie logowania.

## ZADANIA DODATKOWE

### Zadanie 1. Ergonomia pracy z komputerem stacjonarnym i laptopem

Zapoznaj się z filmem instruktażowym opracowanym przez Centralny Instytut Ochrony Pracy na temat optymalnie zaprojektowanego stanowiska pracy przy komputerze (**https://youtu.be/wz0PzZ1FXn4**) i oceń stanowisko komputerowe, na którym pracujesz w domu.




### Zadanie 2. Przepis na zapamiętywanie haseł

Zapoznaj się z poradnikiem *Czy złamanie hasła jest trudne? Jak się zabezpieczyć?* przygotowanym przez CERT Orange Polska (**https://cert.orange.pl/raporty**), a następnie wymyśl i opisz algorytm zapamiętywania hasła, który możesz polecić innym.




### Zadanie 3. Bezpieczeństwo danych z Akademią Khana

Otwórz kurs *Podstawy informatyki – program rozszerzony* w Akademii Khana (**https://pl.khanacademy.org**) i znajdź temat *Bezpieczeństwo danych w sieci*. Zapoznaj się z lekcjami: *Wprowadzenie do ochrony danych w sieci*, *Dane osobowe*, *Śledzenie danych użytkownika* oraz *Cyberataki*, a następnie sprawdź swoją wiedzę.



# ARKUSZ KALKULACYJNY

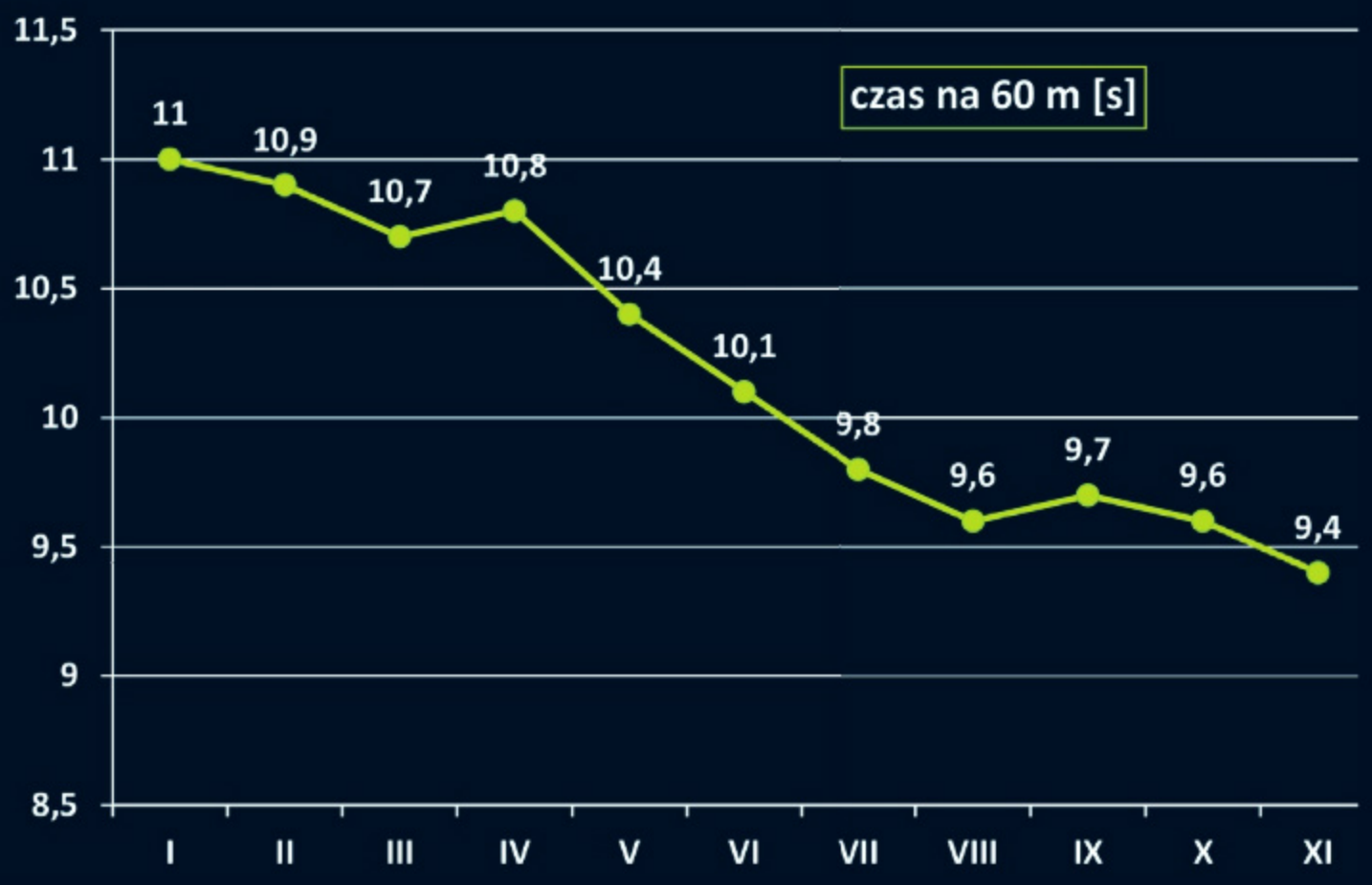
Arkusz kalkulacyjny to bardzo przydatne narzędzie nie tylko w rękach księgowego czy analityka biznesowego. Wystarczy znajomość podstaw, aby efektywnie wykorzystywać arkusz na co dzień:

- do rozwiązywania zadań z matematyki i fizyki,
- do planowania powtórzeń lub nauki języka obcego,
- do kontrolowania wydatków i oszczędności,
- do organizowania zakupów grupowych,
- do śledzenia postępów treningowych.

# ZASTOSOWANIA ARKUSZA KALKULACYJNEGO

**Najczęściej arkusz kalkulacyjny jest postrzegany jako zaawansowany kalkulator pozwalający na sporządzanie diagramów i wykresów, narzędzie umożliwiające opracowywanie analiz, kalkulacji i raportów albo rozwiązywanie abstrakcyjnych problemów algorytmicznych. Tę aplikację warto wykorzystywać na co dzień. Oto cztery praktyczne zastosowania arkusza kalkulacyjnego.**






## POSTĘPY

Arkusz kalkulacyjny pomaga monitorować postępy, np. w bieganiu, pływaniu czy jeździe na rowerze, i skuteczniej planować kolejne treningi. Dane uzupełniane samodzielnie lub wyeksportowane z aplikacji można opracować w formie graficznej.



## BUDŻET

Arkusz kalkulacyjny pozwala zaplanować i kontrolować budżet – wpływy, wydatki i oszczędności (np. na wakacje) w podziale na miesiące, tygodnie czy dni. Wystarczy odpowiednio sformatować tabelę i skorzystać z opcji Σ Autosumowanie.

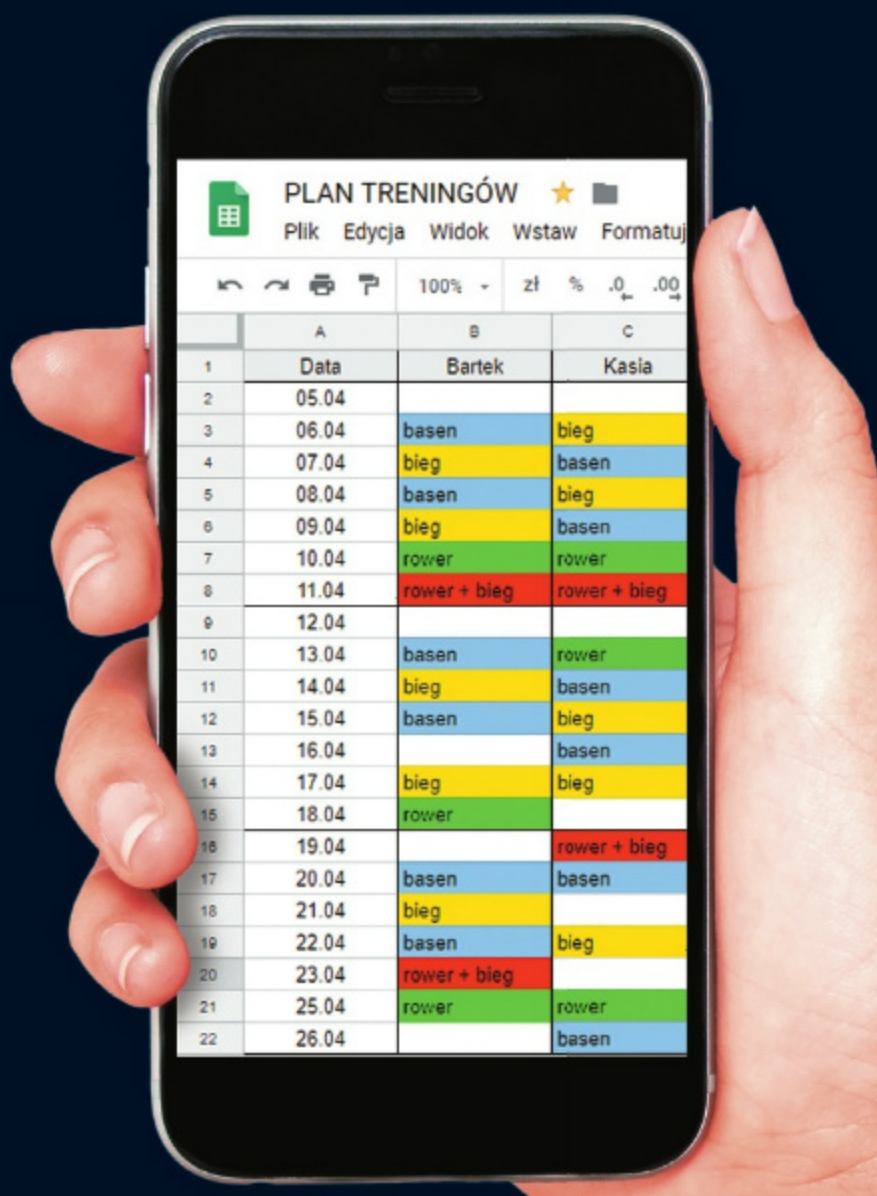
| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | Miesiące | I | II | III | IV | V |
| 2 | **DOCHODY** | | | | | |
| 3 | kieszonkowe | 50,00 | 50,00 | 50,00 | 50,00 | 50,00 |
| 4 | wyprowadzanie psów | 30,00 | 240,00 | 240,00 | 240,00 | 40,00 |
| 5 | stypendia | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |
| 6 | inne | 100,00 | 0,00 | 0,00 | 0,00 | 0,00 |
| 7 | RAZEM | 180,00 | 290,00 | 290,00 | 290,00 | 90,00 |
| 8 | **WYDATKI STAŁE** | | | | | |
| 9 | karta do telefonu | 25,00 | 25,00 | 25,00 | 25,00 | 25,00 |
| 10 | karta do kina | 46,00 | 46,00 | 46,00 | 46,00 | 46,00 |
| 11 | RAZEM | 71,00 | 71,00 | 71,00 | 71,00 | 71,00 |
| 12 | **WYDATKI OKAZJONALNE** | | | | | |
| 13 | słodycze/napoje | 25,00 | 15,00 | 35,00 | 28,00 | 25,00 |
| 14 | kosmetyki | 30,00 | 10,00 | 0,00 | 23,50 | 0,00 |
| 15 | książki/gry/aplikacje | 0,00 | 47,00 | 0,00 | 0,00 | 0,00 |
| 16 | rozrywka | 5,00 | 17,00 | 130,00 | 0,00 | 0,00 |
| 17 | prezenty | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |
| 18 | inne | 0,00 | 0,00 | 12,00 | 0,00 | 18,00 |
| 19 | RAZEM | 60,00 | 89,00 | 177,00 | 51,50 | 43,00 |

# HARMONOGRAMY

Arkusz kalkulacyjny pozwala uporządkować proces przygotowań do egzaminów językowych typu Certificate in Advanced English. Aby skutecznie zaplanować powtórki do części pisemnych i ustnych, wystarczą dobrze sformatowana tabela i kilka formuł warunkowych.

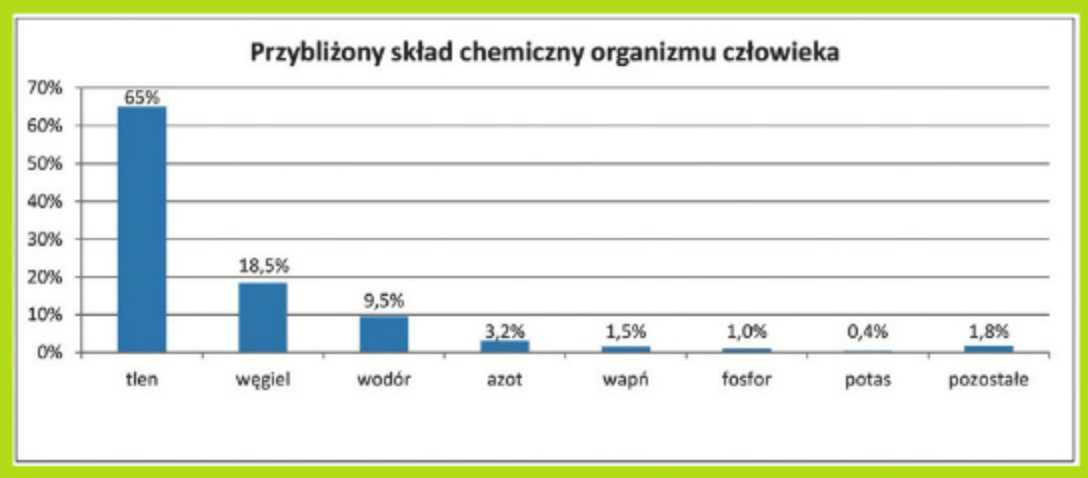
| | A | B | C | D |
|---|---|---|---|---|
| 1 | **Co ile dni** | **1** | **2** | **3** |
| 2 | **Część egzaminu** | **Speaking** | **Use of English** | **Reading** |
| 3 | 1 lutego | TAK | | |
| 4 | 2 lutego | TAK | TAK | |
| 5 | 3 lutego | TAK | | TAK |
| 6 | 4 lutego | TAK | TAK | |
| 7 | 5 lutego | TAK | | |
| 8 | 6 lutego | TAK | TAK | TAK |
| 9 | 7 lutego | TAK | | |
| 10 | 8 lutego | TAK | TAK | |
| 11 | 9 lutego | TAK | | TAK |
| 12 | 10 lutego | TAK | TAK | |
| 13 | 11 lutego | TAK | | |
| 14 | 12 lutego | TAK | TAK | TAK |
| 15 | 13 lutego | TAK | | |
| 16 | 14 lutego | TAK | TAK | |
| 17 | 15 lutego | TAK | | TAK |
| 18 | 16 lutego | TAK | TAK | |
| 19 | 17 lutego | TAK | | |
| 20 | 18 lutego | TAK | TAK | TAK |
| 21 | 19 lutego | TAK | | |

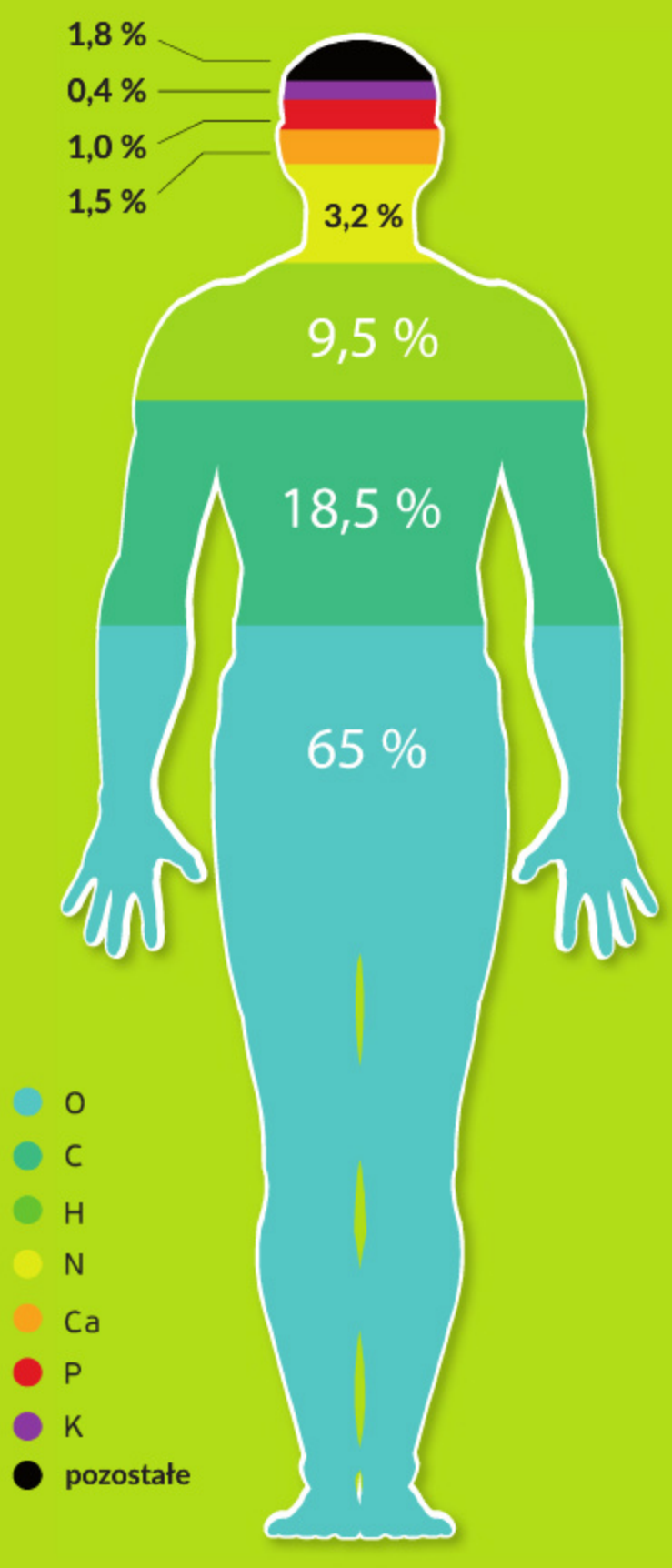
Tabele tworzone w Arkuszach Google i udostępniane w chmurze mogą służyć do ustalania i aktualizowania działalności szkolnego koła wolontariatu, realizacji zespołowego projektu czy wspólnych przygotowań do triathlonu.




# INFOGRAFIKI

Podczas prezentacji referatu w klasie warto zamiast tradycyjnego wykresu słupkowego czy kolumnowego pokazać przygotowaną w arkuszu kalkulacyjnym infografikę, która przyciągnie uwagę kolegów i nauczyciela.




## PRACA Z INFOGRAFIKĄ

1. Jakie rodzaje danych wprowadzono do arkuszy widocznych na infografice?
2. Które z podanych zastosowań arkusza wydają ci się najbliższe? Z których korzystasz?



# 2. Podstawy pracy z arkuszem

- Formatowanie danych
- Wprowadzanie formuł
- Tworzenie wykresów

## ZADANIE NA START

### Ile masz dni

Oblicz, ile masz dni – od bieżącej daty odejmij swoją datę urodzenia. Uwzględnij lata przestępne, tj. lata, które są podzielne przez 4, ale nie są podzielne przez 100, lub są podzielne przez 400. Napisz wzór, który pozwala obliczyć, ile dni upłynęło od 1 stycznia 2000 roku do dzisiaj.

| | A | B | C | D | E | F | G | H | I | J | K | L |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | **Lata przestępne** | | | | | | | | | | | |
| 2 | 1900 | 1911 | 1922 | 1933 | 1944 | 1955 | 1966 | 1977 | 1988 | 1999 | 2010 | 2021 |
| 3 | 1901 | 1912 | 1923 | 1934 | 1945 | 1956 | 1967 | 1978 | 1989 | 2000 | 2011 | 2022 |
| 4 | 1902 | 1913 | 1924 | 1935 | 1946 | 1957 | 1968 | 1979 | 1990 | 2001 | 2012 | 2023 |
| 5 | 1903 | 1914 | 1925 | 1936 | 1947 | 1958 | 1969 | 1980 | 1991 | 2002 | 2013 | 2024 |
| 6 | 1904 | 1915 | 1926 | 1937 | 1948 | 1959 | 1970 | 1981 | 1992 | 2003 | 2014 | 2025 |
| 7 | 1905 | 1916 | 1927 | 1938 | 1949 | 1960 | 1971 | 1982 | 1993 | 2004 | 2015 | 2026 |
| 8 | 1906 | 1917 | 1928 | 1939 | 1950 | 1961 | 1972 | 1983 | 1994 | 2005 | 2016 | 2027 |
| 9 | 1907 | 1918 | 1929 | 1940 | 1951 | 1962 | 1973 | 1984 | 1995 | 2006 | 2017 | 2028 |
| 10 | 1908 | 1919 | 1930 | 1941 | 1952 | 1963 | 1974 | 1985 | 1996 | 2007 | 2018 | 2029 |
| 11 | 1909 | 1920 | 1931 | 1942 | 1953 | 1964 | 1975 | 1986 | 1997 | 2008 | 2019 | 2030 |
| 12 | 1910 | 1921 | 1932 | 1943 | 1954 | 1965 | 1976 | 1987 | 1998 | 2009 | 2020 | 2031 |

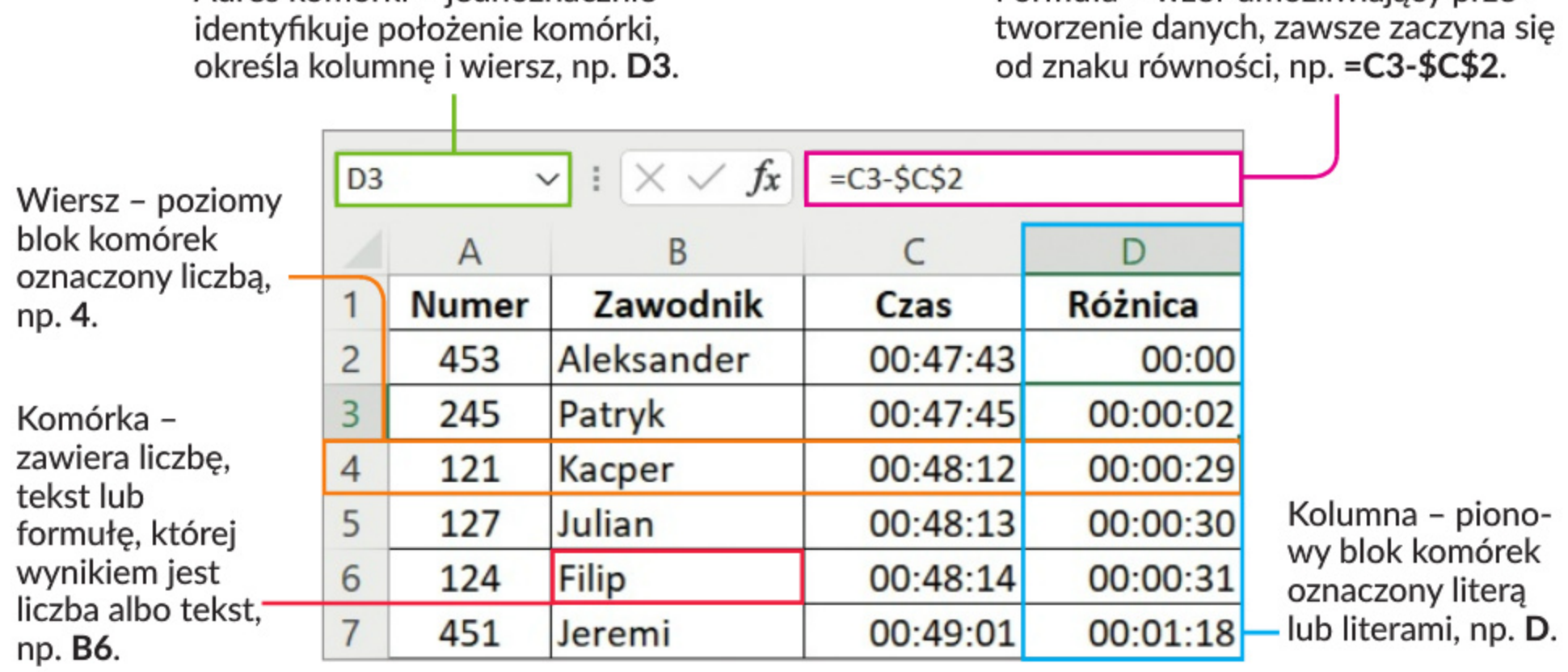
## ARKUSZ KALKULACYJNY

Arkusz kalkulacyjny to wygodne narzędzie analityczne, umożliwiające zbieranie i analizowanie danych, mniej i bardziej skomplikowane obliczenia oraz graficzną prezentację danych. Świetnie sprawdza się też w różnych symulacjach, gdyż po jednorazowym wprowadzeniu danych kolejne modyfikacje nie wymagają wpisywania ich od początku, tylko uwzględnienia zmian.

Każdy dokument arkusza kalkulacyjnego (skoroszyt, zeszyt) składa się z zestawu dwuwymiarowych tabel oznaczanych w poziomie wielkimi literami, a w pionie – liczbami. Pozwala to na wygodną obróbkę danych oraz ich prezentację zarówno w formie tabelarycznej, jak i w graficznej (wykresy).

Do najpopularniejszych arkuszy kalkulacyjnych należą Microsoft Excel, Apache OpenOffice Calc, LibreOffice Calc, Apple Numbers oraz Arkusze Google. Chociaż różnią się one między sobą, poznanie zasady działania jednego programu powoduje, że bez problemu można obsługiwać inny, zwłaszcza że twórcy starają się, aby podstawowe funkcje miały te same nazwy.

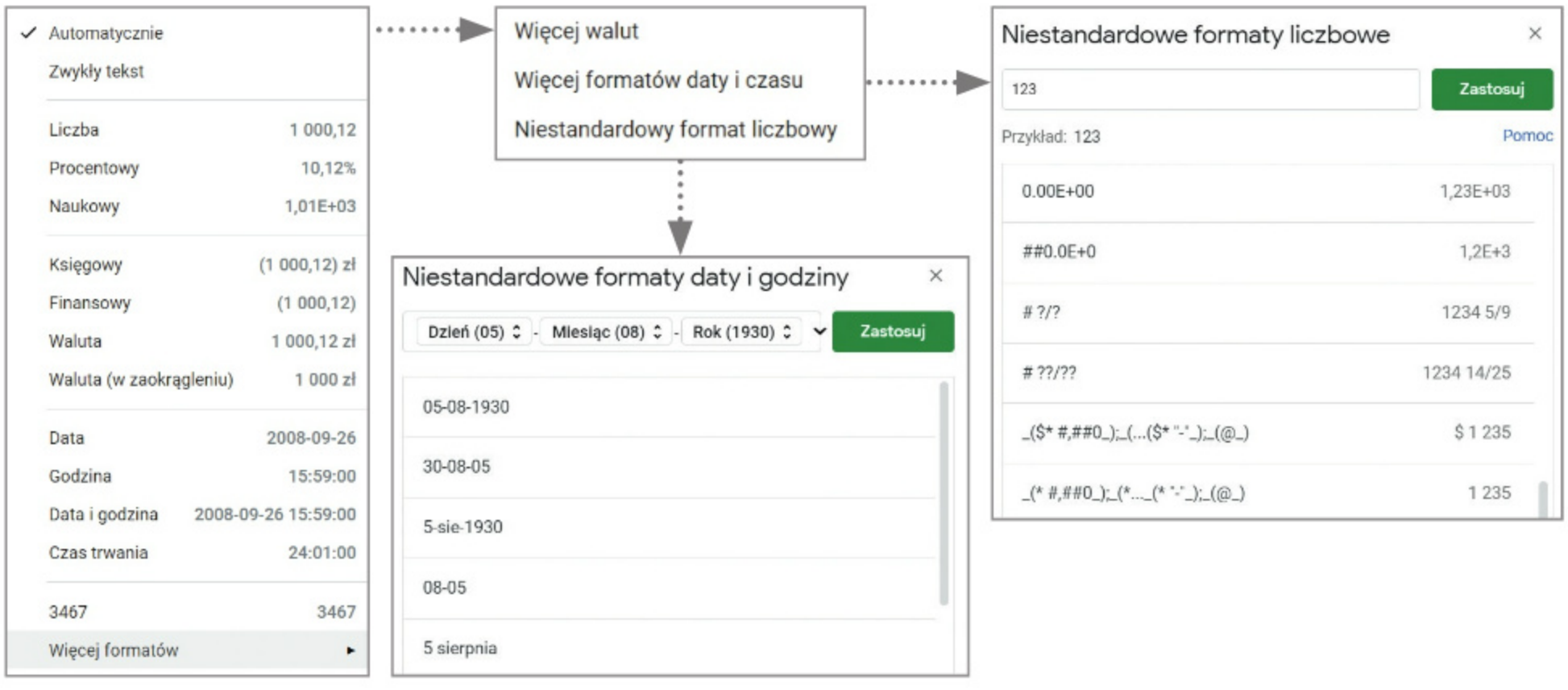
Obszar roboczy arkusza przedstawiony na rys. 1 zawiera tabelę z danymi i formułami, za pomocą których można obliczyć różnice czasów osiągniętych przez zawodników.




**Rys. 1.** Podstawowe elementy arkusza kalkulacyjnego na przykładzie aplikacji Excel

## FORMATY DANYCH

Do komórek arkusza wpisuje się dane w postaci tekstowej (na rys. 1 kolumna **B**) i liczbowej (np. kolumna **A**), przy czym liczby można przedstawiać np. w formie dat, procentów czy walut albo numerów specjalnych typu PESEL – w zależności od potrzeb.



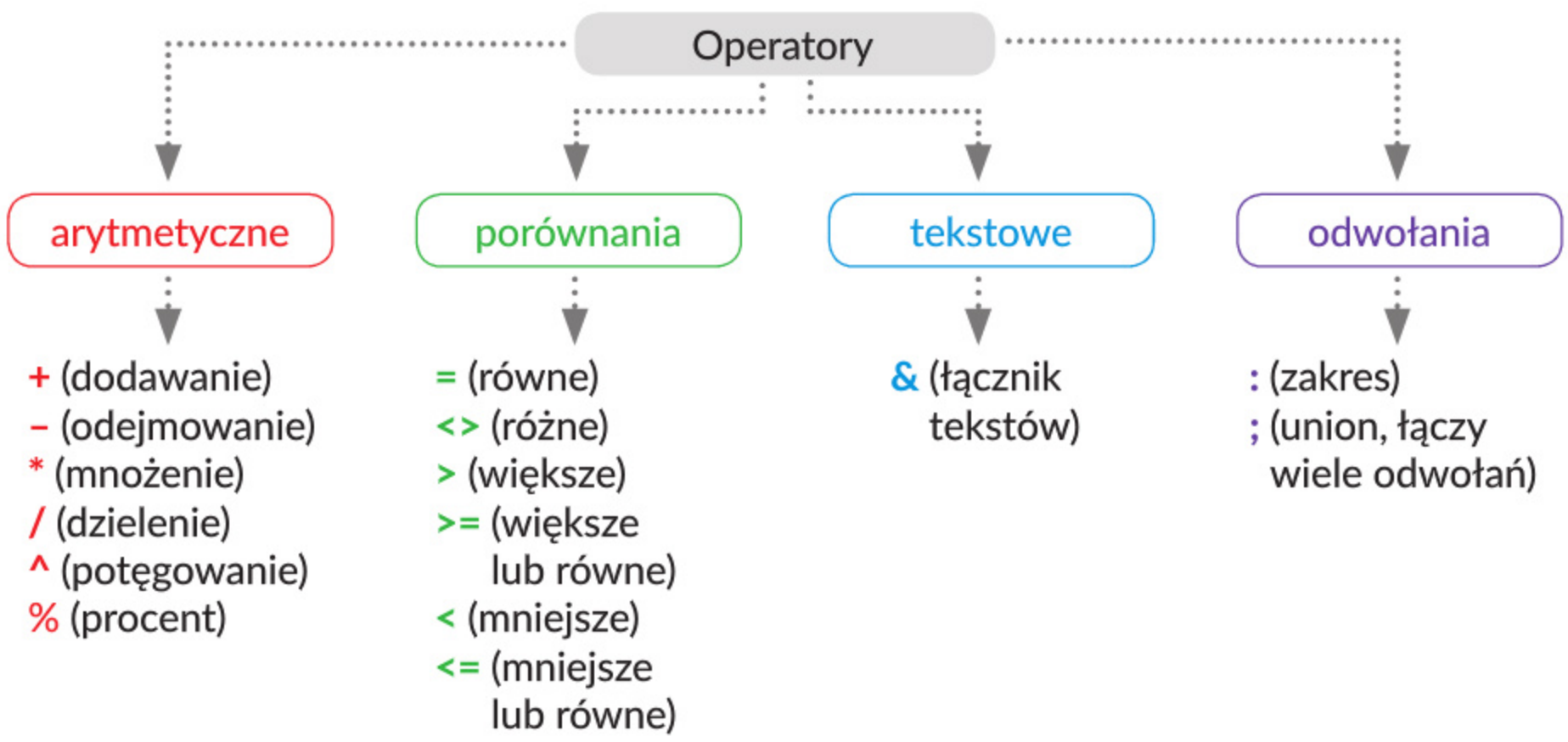

**Rys. 2.** Formaty zapisu liczb w aplikacji Arkusze Google

**W PRAKTYCE**

Data może być różnie wyświetlana (np. 24 kwi 86, 1986-04-24, 24 kwiecień 1986) i wykorzystywana w obliczeniach, ponieważ arkusz kalkulacyjny przechowuje ją w postaci liczby dni między datą wpisaną a ustaloną jako zerowa. Domyślnym dniem zerowym (liczbą 1) dla programu Excel jest 1 stycznia 1900 roku, co oznacza, że data 1 stycznia 2018 roku jest liczbą 43 101.

## WPROWADZANIE FORMUŁ

Oprócz danych tekstowych i liczbowych w arkuszu wpisuje się formuły. Wzory, które poprzedza się zawsze znakiem równości, mogą składać się z operatorów: arytmetycznych, porównań i tekstowych oraz stałych w postaci liczb lub tekstu ograniczonego znakami " ", zmiennych w postaci adresów komórek lub zakresów komórek, wartości logicznych **PRAWDA** i **FAŁSZ**, funkcji wbudowanych, których argumentami mogą być liczby, tekst, wartości logiczne, odwołania do komórek oraz inne funkcje. Obliczenia są wykonywane zgodnie z zasadą kolejności działań. W razie potrzeby należy używać nawiasów okrągłych ( ).



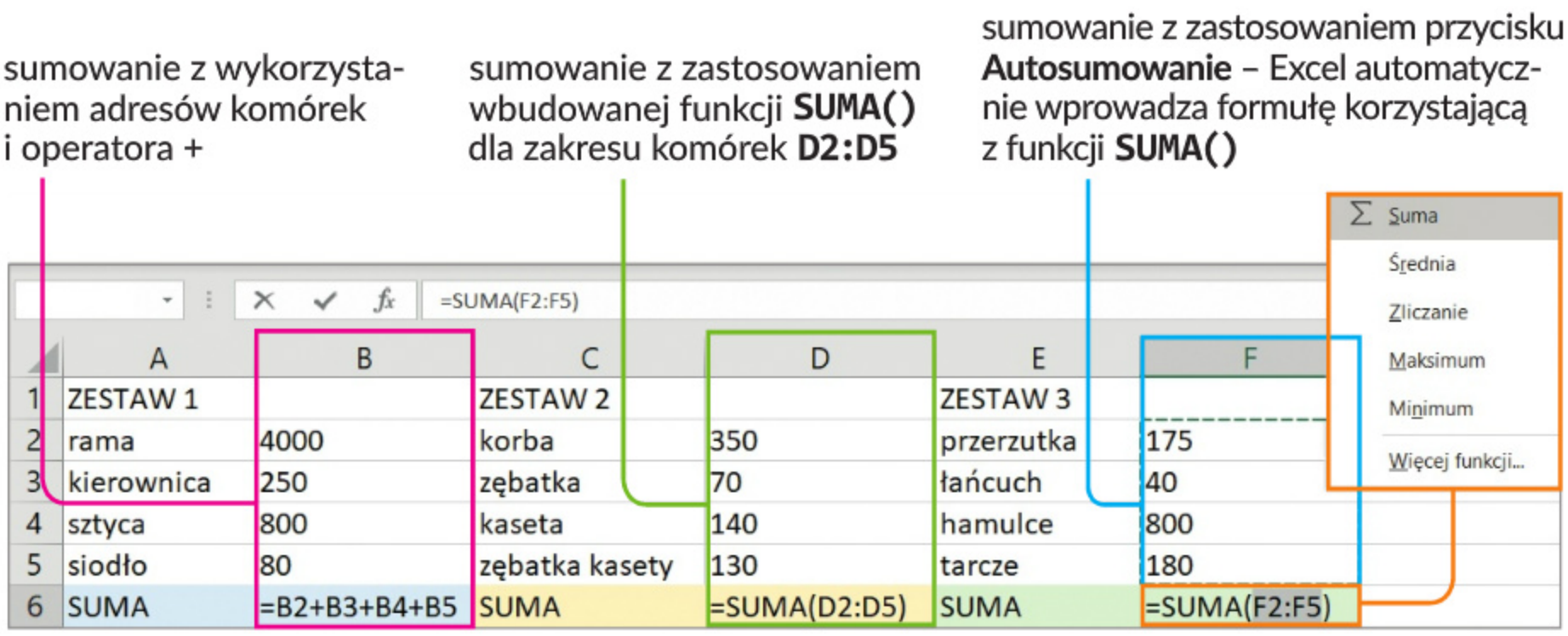

**Rys. 3.** Operatory wykorzystywane w arkuszu Excela

**W PRAKTYCE**

Aby w Excelu przełączać się między trybami wyświetlania wyniku formuły (widok normalny) i wyświetlania formuł, należy zastosować kombinację klawiszy **Ctrl+`** lub na karcie **Formuły** w grupie **Inspekcja formuł** kliknąć polecenie **Pokaż formuły**.

Funkcje są pogrupowane w różne kategorie.

- Na liście funkcji matematycznych i trygonometrycznych znajdują się m.in. **SUMA** (dodaje liczby w podanym zakresie komórek) i **PI** (wartość liczby π z dokładnością do 15 cyfr po przecinku).



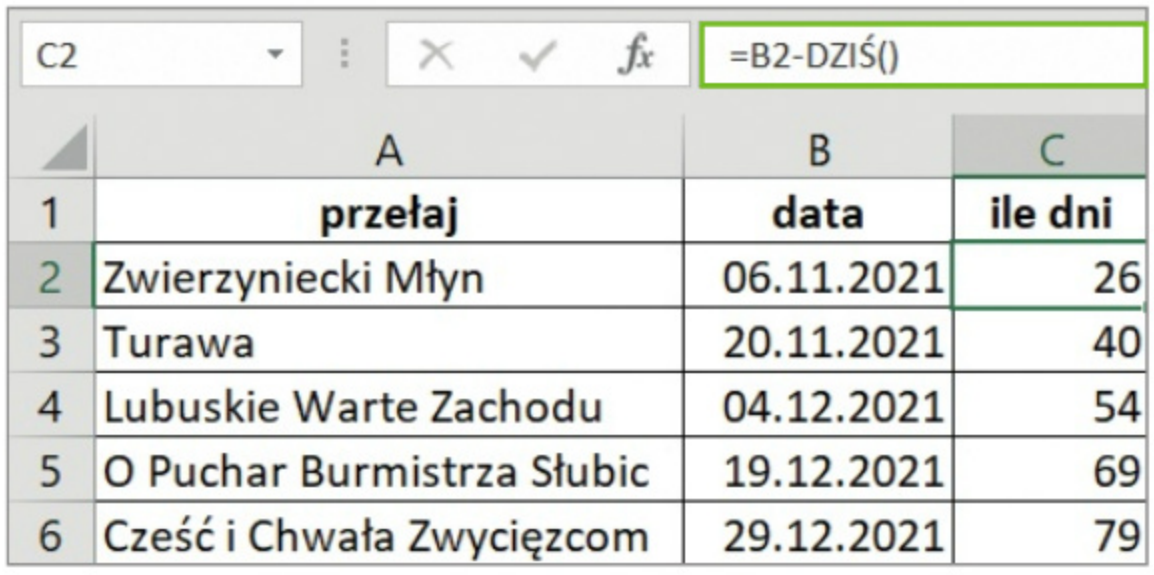

**Rys. 4.** Wycena używanych części rowerowych – zastosowanie operatorów i funkcji w arkuszu Excela

- Do funkcji statystycznych należą **MAX** (wyznacza największą wartość z danego zbioru), **MIN** (wyznacza najmniejszą wartość z danego zbioru) i **ŚREDNIA** (wyznacza średnią arytmetyczną podanych argumentów).

| | A | B | C | D |
|---|---|---|---|---|
| 1 | **Średnie ceny brutto sprawdzania geometrii ramy rowerowej w woj. opolskim** | | | |
| 2 | Brzeg | 141,12 | | |
| 3 | Głubczyce | 147,06 | min= | =MIN(B2:B10) |
| 4 | Kędzierzyn-Koźle | 132,30 | max= | =MAX(B2:B10) |
| 5 | Kluczbork | 144,43 | średnia= | =ŚREDNIA(B2:B10) |
| 6 | Krapkowice | 137,53 | | |
| 7 | Namysłów | 135,92 | | |
| 8 | Nysa | 132,30 | | |
| 9 | Olesno | 151,87 | | |
| 10 | Opole | 143,32 | | |

**Rys. 5.** Przykład zastosowania funkcji statystycznych w arkuszu Excela

- W grupie z datami i godzinami można znaleźć np. funkcje **DZIŚ** (wskazuje bieżącą datę) i **TERAZ** (wskazuje bieżącą datę i godzinę). Wyświetlane wartości są aktualizowane przy każdym otwarciu arkusza, wprowadzanych zmianach lub wymuszeniu przeliczenia na żądanie.




| | A | B | C |
|---|---|---|---|
| 1 | **przełaj** | **data** | **ile dni** |
| 2 | Zwierzyniecki Młyn | 06.11.2021 | 26 |
| 3 | Turawa | 20.11.2021 | 40 |
| 4 | Lubuskie Warte Zachodu | 04.12.2021 | 54 |
| 5 | O Puchar Burmistrza Słubic | 19.12.2021 | 69 |
| 6 | Cześć i Chwała Zwycięzcom | 29.12.2021 | 79 |

**Rys. 6.** Przykład zastosowania funkcji **DZIŚ()** w arkuszu Excela

Po wstawieniu znaku równości nazwę funkcji można wpisać do formuły samodzielnie lub wyszukać w danej kategorii. Z arkuszem Excela można postąpić dwojako:
Sposób 1.

- kliknąć kartę **Formuły**,
- w grupie **Biblioteka funkcji** wybrać odpowiednią kategorię,
- rozwinąć listę i kliknąć kursorem nazwę potrzebnej funkcji.

Sposób 2.

- kliknąć przycisk ***fx*** przy pasku formuły, co umożliwia przejrzenie listy wszystkich funkcji,
- zaznaczyć funkcję kliknięciem i zatwierdzić jej wybór przyciskiem **OK**.

W arkuszu LibreOffice Calc trzeba wybrać kartę **Wstaw** i opcję **Funkcja**, za pomocą której otwiera się okno **Kreatora funkcji**, albo skorzystać z przycisku ***fx*** przy pasku formuły. Użytkownicy Arkuszy Google mogą wybrać kartę **Wstaw** i opcję **Funkcja** albo na pasku menu kliknąć przycisk **Σ** i rozwinąć listę dostępnych funkcji.

Każdą formułę można skopiować – należy zaznaczyć komórkę, w którą została wpisana, a następnie przeciągnąć ją odpowiednio w bok, w pionie lub po skosie za pomocą uchwytu wypełnienia. Zakres wskazany w ten sposób zostanie automatycznie wypełniony.

D3 | =C3-$C$2

| | A | B | C | D |
|---|---|---|---|---|
| 1 | **Numer** | **Zawodnik** | **Czas** | **Różnica** |
| 2 | 453 | Aleksander | 00:47:43 | 00:00 |
| 3 | 245 | Patryk | 00:47:45 | 00:00:02 |
| 4 | 121 | Kacper | 00:48:12 | |
| 5 | 127 | Julian | 00:48:13 | |
| 6 | 124 | Filip | 00:48:14 | |
| 7 | 451 | Jeremi | 00:49:01 | |

**Rys. 7.** Kopiowanie formuły obliczania straty do zwycięzcy wyścigu

W przypadku arkusza na rys. 7, zawierającego wyniki wyścigów rowerowych kolarzy górskich, do komórki **D3** wpisano formułę pozwalającą obliczyć stratę zawodnika, który zajął drugie miejsce, tj. **=C3-$C$2**. Od czasu osiągniętego przez zawodnika numer 245 (wartość w komórce **C3**) odjęto czas zwycięzcy (wartość w komórce **C2**). Zastosowano odpowiednio adresowanie względne i bezwzględne.

## ADRESOWANIE KOMÓREK

W trakcie wprowadzania formuły należy wziąć pod uwagę, czy wartości mają się zmieniać podczas kopiowania formuły do innych komórek, czy też nie, i zastosować – odpowiednio – adres względny, bezwzględny lub mieszany.

- **Adres względny** komórki ma postać: litera (ciąg liter) i liczba np. **B11**. Stosuje się go w celu automatycznej zamiany adresów podczas przeciągania formuł.

- **Adres bezwzględny** komórki zawiera znak dolara zarówno przed literą, jak i przed liczbą, np. **$B$11**. Używa się go wtedy, gdy podczas kopiowania formuł nie powinna się zmienić wartość danej komórki.
- **Adres mieszany** komórki stanowi połączenie adresów bezwzględnego i względnego, np. **$B11** albo **B$11**. Jest on przydatny, gdy określone wartości mają się odnosić do konkretnej kolumny lub konkretnego wiersza.

| C | D |
|---|---|
| **Czas** | **Różnica** |
| 0,033136574074074 1 | 0 |
| 0,033159722222222 | =C3-$C$2 |
| 0,033472222222222 | =C4-$C$2 |
| 0,033483796296296 3 | =C5-$C$2 |
| 0,033495370370370 4 | =C6-$C$2 |
| 0,034039351851851 9 | =C7-$C$2 |

adresowanie względne (podczas kopiowania formuły adres komórki zawierającej czas osiągnięty przez zawodnika, który nie zajął pierwszego miejsca, zmienia się)

adresowanie bezwzględne (podczas kopiowania formuły adres komórki zawierającej czas zwycięzcy pozostaje bez zmian)

**Rys. 8.** Przykład adresowania komórek w tabeli z rys. 7

## Ćwiczenie 1. Obliczanie dni

Przygotuj tabelę, za pomocą której można obliczyć wiek zawodników w dniach (wzór poniżej). Uzupełnij kolumnę z nagłówkiem **Ile dni**.

| | A | B | C |
|---|---|---|---|
| 1 | **Kolarz** | **Data urodzenia** | **Ile dni** |
| 2 | Dariusz Batek | 27 kwi 86 | |
| 3 | Piotr Brzózka | 14 paź 89 | |
| 4 | Aleksandra Dawidowicz | 4 lut 87 | |
| 5 | Adrian Działakiewicz | 23 lut 88 | |
| 6 | Piotr Formicki | 31 maj 82 | |

1. Zapisz odpowiednią formułę w komórce **C2** – od bieżącej daty wskazanej przez funkcję **DZIŚ** odejmij datę urodzenia zawodnika. Ponieważ data urodzenia w formule musi się odpowiednio zmieniać dla każdego kolarza, zastosuj adresowanie względne.
   `=DZIŚ()-B2`
2. Skopiuj formułę z komórki **C2** do pozostałych komórek.

| | A | B | C |
|---|---|---|---|
| 1 | **Kolarz** | **Data urodzenia** | **Ile dni** |
| 2 | Dariusz Batek | 31529 | =DZIŚ()-B2 |
| 3 | Piotr Brzózka | 32795 | =DZIŚ()-B3 |
| 4 | Aleksandra Dawidowicz | 31812 | =DZIŚ()-B4 |
| 5 | Adrian Działakiewicz | 32196 | =DZIŚ()-B5 |
| 6 | Piotr Formicki | 30102 | =DZIŚ()-B6 |

## Ćwiczenie 2. Obliczanie parametrów

Przygotuj tabelę, za pomocą której można obliczyć średnią prędkość rowerzysty, pożądaną wysokość ramy w rowerze górskim oraz promień i obwód koła (wzór poniżej). Uzupełnij kolumnę z nagłówkiem **Obliczenia**.

| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | | | | | **Obliczenia** |
| 2 | droga [km] | 10 | | średnia prędkość [km/h] | |
| 3 | czas [min] | 30 | | | |
| 4 | | | | | |
| 5 | długość nogi [cm] | 80 | | wysokość ramy [cm] | |
| 6 | współczynnik | 0,57 | | wysokość ramy [cal] | |
| 7 | | | | | |
| 8 | średnica koła [cal] | 28 | | promień koła [cm] | |
| 9 | | | | obwód koła [mm] | |

1. Wyznacz w komórce **E2** średnią prędkość i uwzględnij zamianę jednostek.
   `=B2/(B3/60)`

2. Wyznacz w komórce **E5** wysokość ramy w centymetrach – do obliczeń wykorzystaj wzór $h_r = h_n \cdot w$.
   `=B5*B6`

3. Wyznacz w komórce **E6** wysokość ramy w calach – wykorzystaj wysokość ramy w centymetrach i uwzględnij zamianę jednostek (1 cal = 2,54 cm).
   `=E5/2,54`

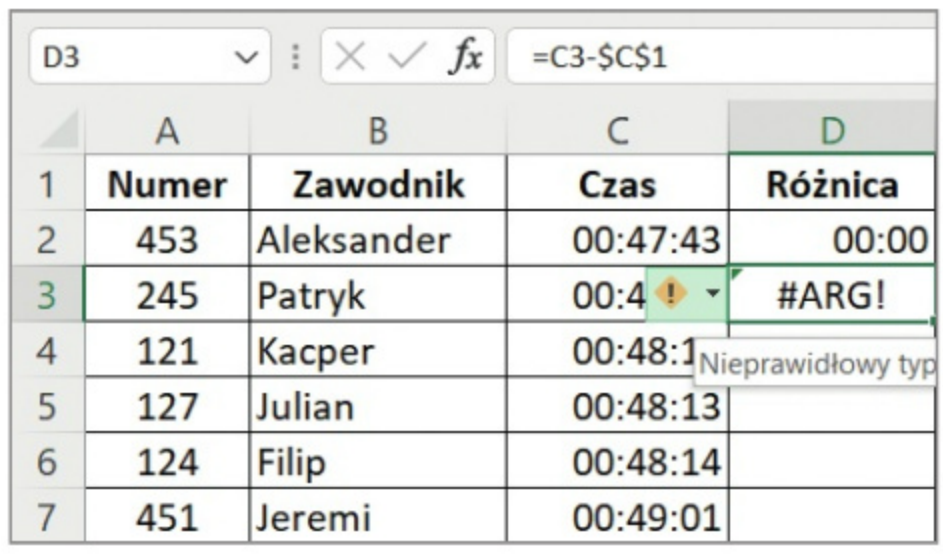
4. Wyznacz w komórce **E8** promień koła – wykorzystaj długość średnicy i uwzględnij zamianę jednostek.
   `=B8/2*2,54`

5. Wyznacz w komórce **E9** obwód koła – wykorzystaj funkcję **PI** oraz promień koła i uwzględnij zamianę jednostek.
   `=2*PI()*E8*10`

6. W razie potrzeby odpowiednio sformatuj wyniki.

### W PRAKTYCE

Jeśli w arkuszu wyświetli się błąd **#ARG!**, to oznacza, że wystąpił problem z zapisem formuły lub komórkami, do których się odwoływano. Ponieważ jest to ogólny błąd, warto skorzystać z pomocy technicznej Excela (**https://support.microsoft.com/pl-PL/excel**), która zawiera rozwiązania typowych problemów.



## Ćwiczenie 3. Obliczanie wyników głosowania

Przygotuj tabelę prezentującą preferencje uczniów, dotyczące zwiedzania podczas rowerowej wycieczki po województwie świętokrzyskim (wzór poniżej). Uzupełnij kolumnę **Procent głosów**.

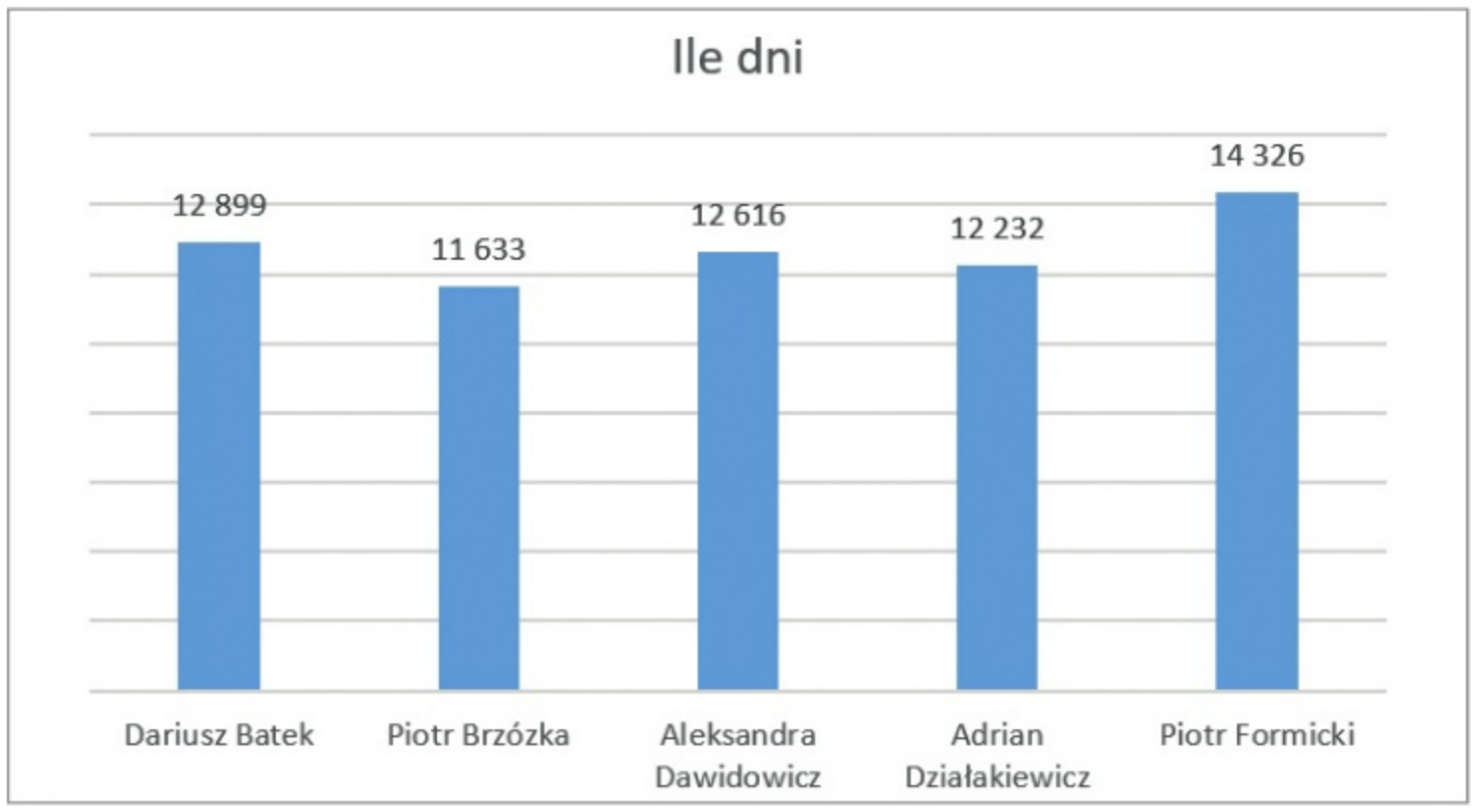
| | A | B | C |
|---|---|---|---|
| 1 | **Co chcemy odwiedzić?** | **Liczba głosów** | **Procent głosów** |
| 2 | Muzeum im. „Orła Białego” w Skarżysku-Kamiennej | 9 | |
| 3 | klasztor opactwa cysterskiego w Wąchocku | 13 | |
| 4 | dąb Bartek w Zagnańsku | 14 | |
| 5 | osada średniowieczna w Hucie Szklanej | 3 | |
| 6 | platforma widokowa nad gołoborzem na Świętym Krzyżu | 21 | |

1. Wyznacz w komórce **B7** ogólną liczbę oddanych głosów.
   `=SUMA(B2:B6)`
2. Zapisz odpowiednią formułę w komórce **C2**. W odniesieniu do ogólnej liczby głosów zastosuj adres bezwzględny, który blokuje zmianę adresu.
   `=B2/$B$7*100%`
3. Skopiuj formułę z komórki **C2** do pozostałych komórek.

### WYKRESY

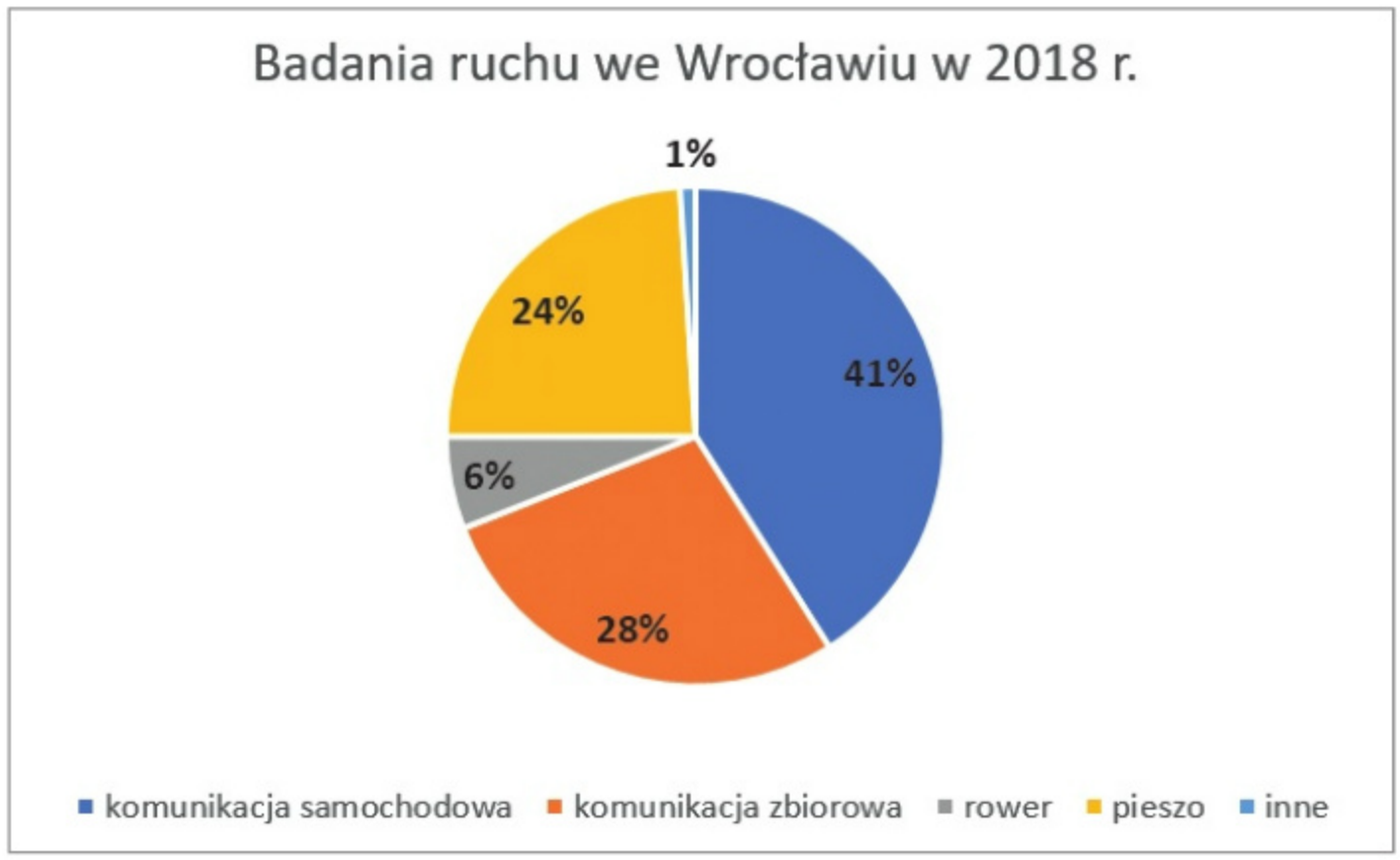
Przedstawienie danych w formie graficznej nierzadko przekłada się na łatwiejsze zrozumienie zawartych w nich informacji oraz relacji zachodzących między poszczególnymi pozycjami. Arkusz kalkulacyjny oferuje wiele typów i podtypów wykresów: każdy przeznaczony do innych celów. W zależności od potrzeb można użyć odpowiedniego typu wykresu.

- Do porównania wartości liczbowych wykorzystuje się wykresy kolumnowe lub słupkowe w wersji dwu- lub trójwymiarowej oraz ich odmiany. W wykresach słupkowych łatwiej dodać czytelne etykiety. W zależności od potrzeb można tworzyć wykresy grupowane (odnoszące się do kilku serii danych) i skumulowane (np. kolumnowy i słupkowy, całościowo reprezentujące sumę wszystkich serii danych).



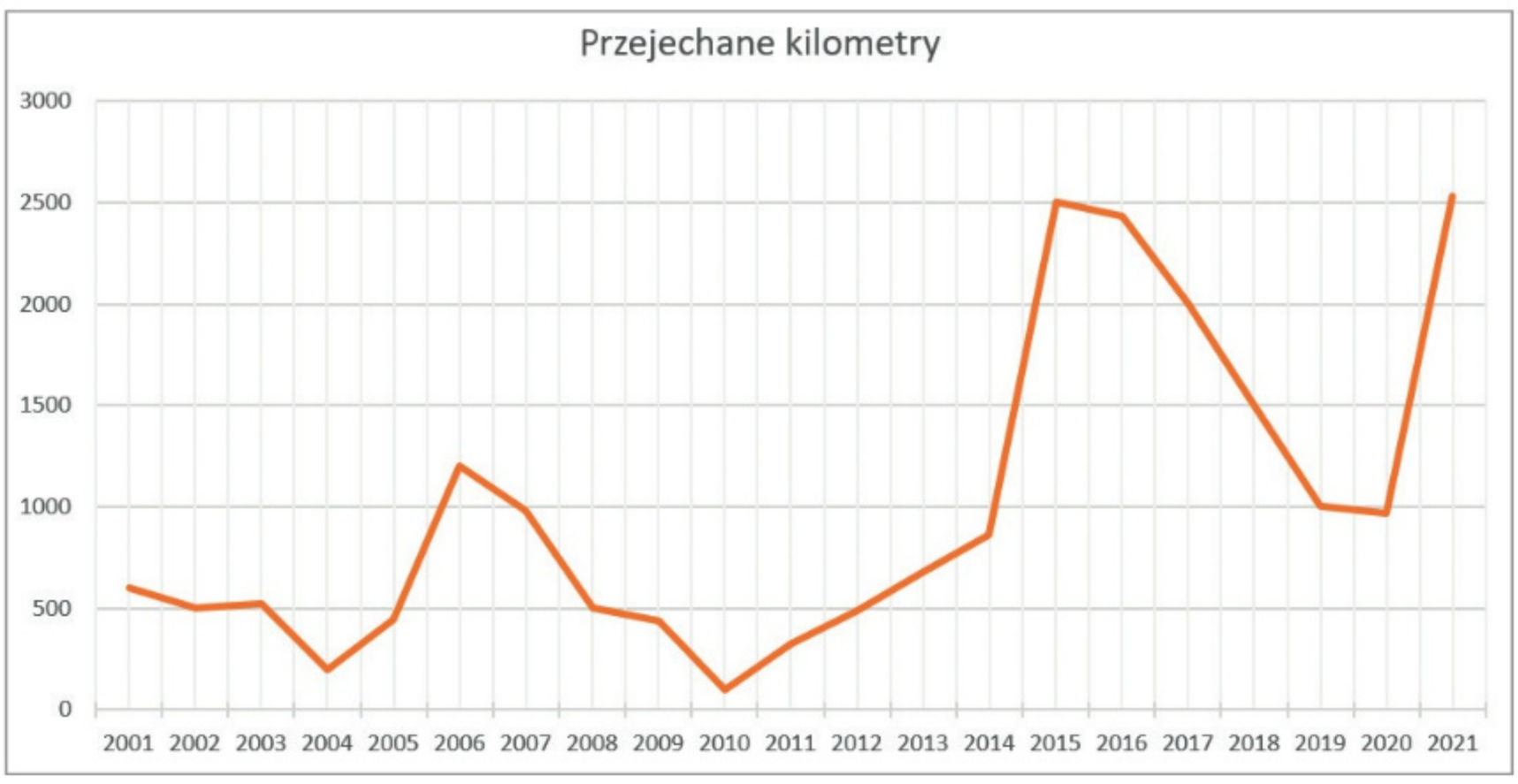

**Rys. 9.** Wykres kolumnowy 2D prezentujący wiek kolarzy w dniach

- Na potrzeby analizy procentowego udziału danej kategorii w całej próbie stosuje się wykresy kołowe lub pierścieniowe oraz ich odmiany.




**Rys. 10.** Wykres kołowy prezentujący udział ruchu rowerowego we Wrocławiu w roku 2018

- Dane zmieniające się w sposób ciągły w danym czasie, w tym tendencje wzrostowe lub spadkowe, najwygodniej obserwuje się na wykresie liniowym. Na jednym rysunku można zaprezentować dane z jednej lub kilku serii.

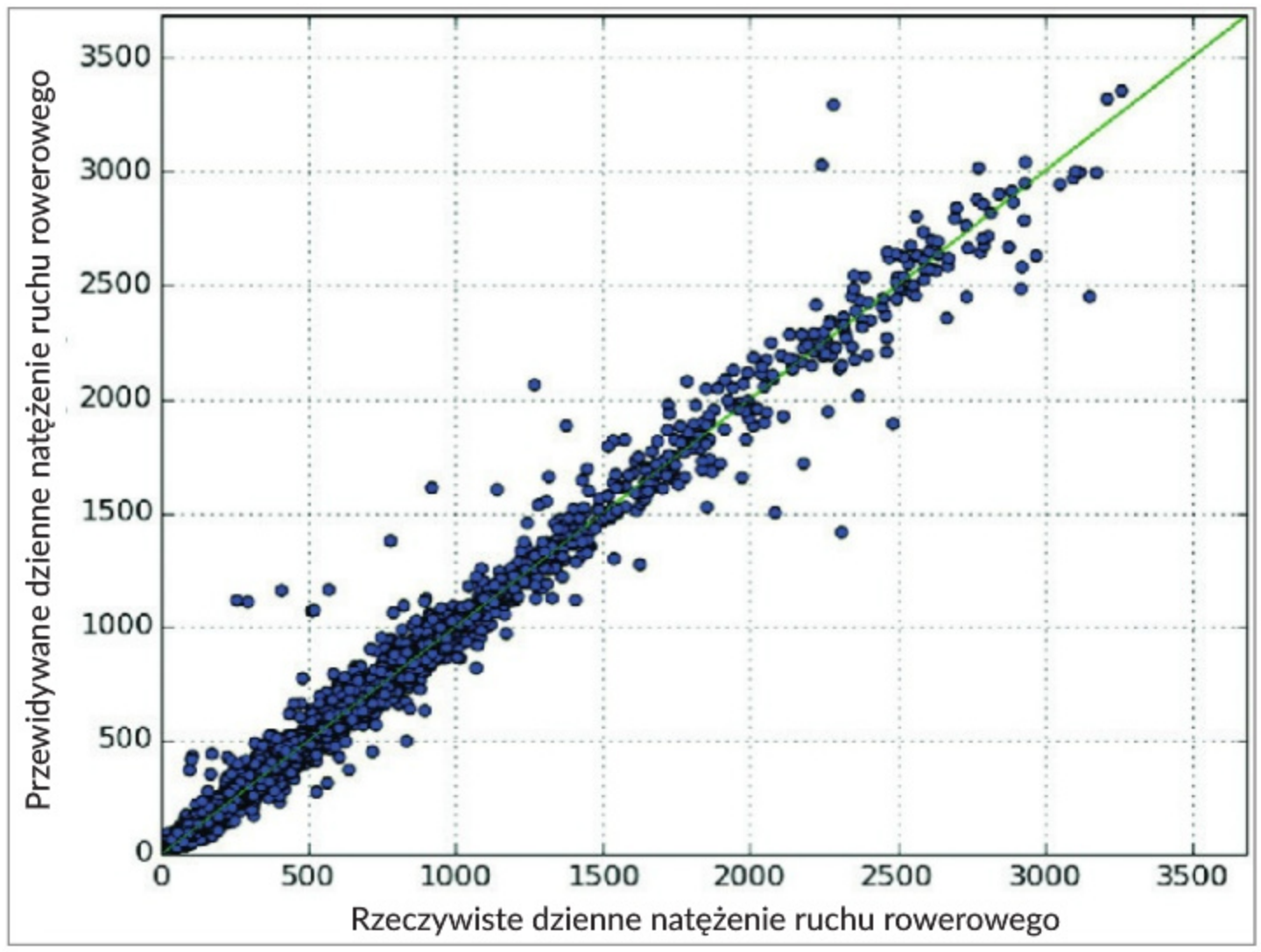



**Rys. 11.** Wykres liniowy przedstawiający aktywność rowerzysty (przejechane kilometry) w latach 2001–2021

**WIEM WIĘCEJ**

Należy pamiętać, że wykresami można manipulować, tak by pokazać odbiorcy dokładnie to, co chcemy. Dobór skali znacząco wpływa na wydźwięk wykresu, a w przypadku analizy dwóch wykresów może wprowadzić w błąd. Warto zapoznać się z filmowymi przykładami zamieszczonymi na stronie Akademii Khana – „Czytanie wykresów liniowych" oraz „Mylące wykresy liniowe".

- Do pokazania zależności dwóch zmiennych w układzie współrzędnych najlepszy jest wykres punktowy (XY).




**Rys. 12.** Wykres punktowy przedstawiający przewidywane i rzeczywiste dzienne natężenie ruchu rowerowego

Aby przygotować wykres w Excelu, trzeba wykonać następujące czynności:

- zaznaczyć serię danych (zakres komórek z wiersza lub kolumny),
- wybrać na karcie **Wstawianie** grupę **Wykresy** i zaznaczyć pożądaną opcję,
- opisać rysunek tak, aby był czytelny – wprowadzić tytuł, dodać albo usunąć etykiety osi kategorii (w przypadku wykresów prezentowanych z wykorzystaniem układu współrzędnych; zwykle są to nagłówki kolumn lub wierszy) oraz etykiety danych (mogą zawierać nazwy serii, kategorii lub wartości),
- zmodyfikować lub usunąć legendę.

Po kliknięciu obszaru wykresu pojawi się karta **Narzędzia wykresów**, za pomocą której można wprowadzić wspomniane zmiany. Korektę można też nanieść po kliknięciu poszczególnych elementów prawym przyciskiem myszy.

**W PRAKTYCE**

Aby umieścić wykres w dokumencie tekstowym, np. w referacie, można:

- przygotować wykres w arkuszu, a następnie wykonać zrzut i wstawić go do tekstu, tak jak każdą inną grafikę;
- w arkuszu przygotować wykres, zaznaczyć go i skopiować do schowka (**Ctrl+C**), a następnie w dokumencie kliknąć odpowiednie miejsce i wkleić (**Ctrl+V**);
- przygotować wykres, np. w Wordzie – na karcie **Wstawianie** należy wybrać grupę **Ilustracje**, kliknąć opcję **Wykres** i wskazać odpowiedni typ wykresu; na stronie pojawi się szablon wykresu, a także okno arkusza, które należy odpowiednio uzupełnić danymi.

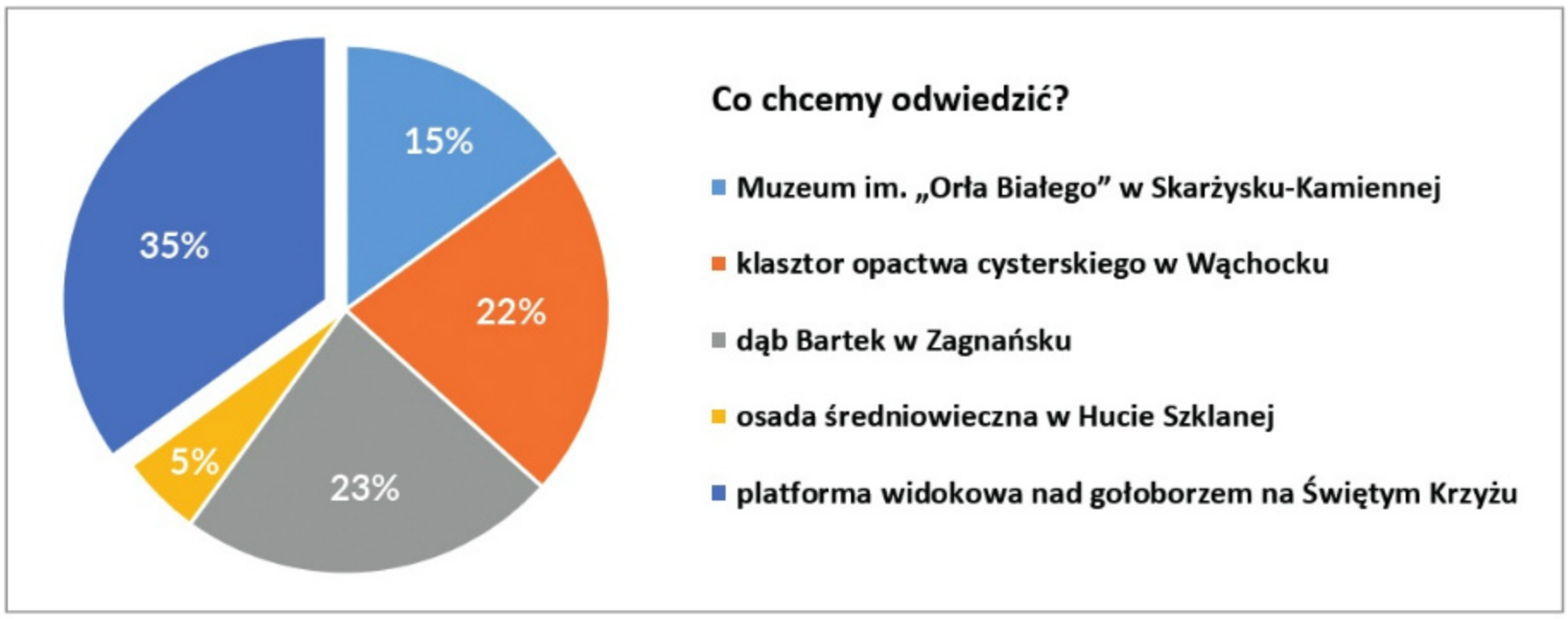
## Ćwiczenie 4. Prezentowanie wyników głosowania

Przedstaw w formie graficznej preferencje uczniów dotyczące zwiedzania województwa świętokrzyskiego. Do wyników uzyskanych w ćwiczeniu 3 dobierz odpowiedni typ wykresu i zadbaj o jego czytelność.

1. Zaznacz zakres danych **C2:C6**.

2. Dobierz wykres – ponieważ tabela zawiera wyniki procentowe, najlepiej zastosować dwuwymiarowy wykres kołowy.
   **Wstawianie → Wykresy → Wstaw wykres kołowy lub pierścieniowy → Kołowy 2D → Kołowy**

3. Opisz wykres – uwzględnij tytuł ankiety, legendę prezentującą możliwe odpowiedzi oraz etykiety danych. Dobierz kolorystykę.




## Ćwiczenie 5. Prezentowanie trasy przejazdu

Wykonaj obliczenia, a następnie przedstaw w formie graficznej profil topograficzny trasy rowerowej z Michniowa do Wąchocka.

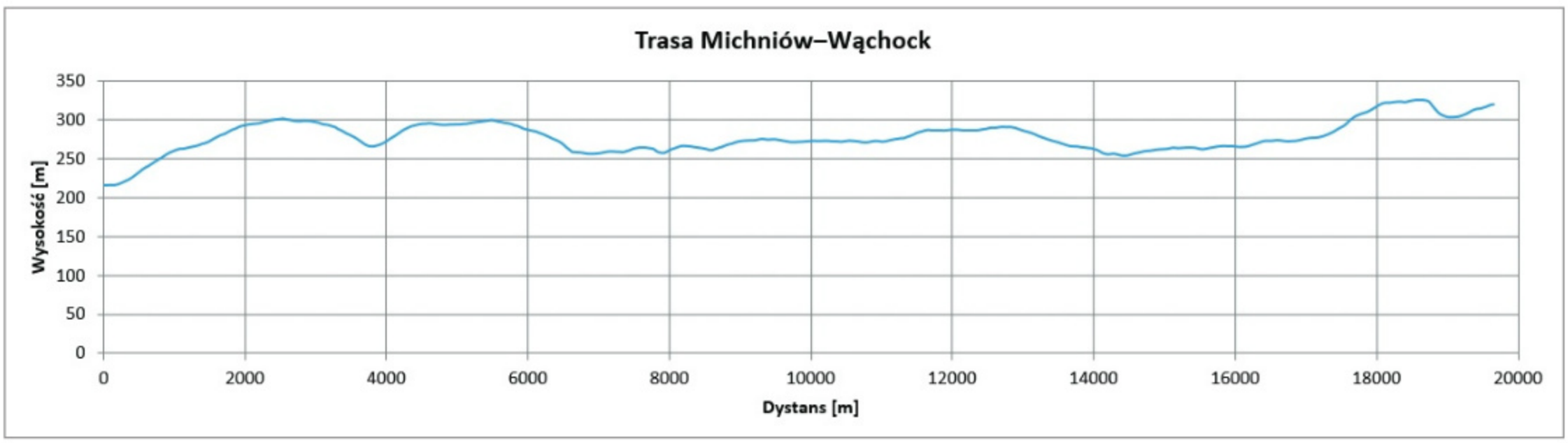
a. Przygotuj tabelę na podstawie danych z serwisu **www.geocontext.org** (wzór poniżej) i wyznacz najwyższy oraz najniższy punkt na trasie, średnią wysokość, a także różnicę wysokości między punktem końcowym a punktem startowym.
b. Sporządź wykres.

| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | **Dystans [m]** | **Wysokość [m]** | | | **[m n.p.m.]** |
| 2 | 0 | 217 | | najwyższy punkt | |
| 3 | 38 | 217 | | najniższy punkt | |
| 4 | 77 | 217 | | średnia wysokość | |
| 5 | 115 | 217 | | różnica wysokości | |
| 6 | 154 | 217 | | | |
| 7 | 192 | 218 | | | |
| 8 | 231 | 220 | | | |
| 9 | 269 | 221 | | | |
| 10 | 308 | 223 | | | |
| 11 | 346 | 224 | | | |

1. Aby wykonać polecenia z punktu **a**, wyznacz szukane wartości za pomocą odpowiednich formuł.

| | D | E |
|---|---|---|
| 1 | | **[m n.p.m.]** |
| 2 | najwyższy punkt | =MAX(B:B) |
| 3 | najniższy punkt | =MIN(B:B) |
| 4 | średnia wysokość | =ŚREDNIA(B:B) |
| 5 | różnica wysokości | =B513-B2 |

2. Aby wykonać polecenia z punktu **b**, określ, jakiego typu wykres najczytelniej prezentuje wartości zmieniające się w sposób ciągły, zaznacz odpowiedni zakres danych, wstaw wykres i go opisz. Porównaj otrzymany wykres z profilem topograficznym trasy stworzonym przez program Geocontext-Profiler.




## Ćwiczenie 6. Prezentowanie uzyskanych wyników

Wykonaj obliczenia, a następnie przedstaw w formie graficznej czasy uzyskane przez uczestniczki zawodów.

**a.** Przygotuj tabelę (wzór poniżej) i oblicz straty czasu w stosunku do najlepszego wyniku oraz średnią prędkość na trasie.

**b.** Sporządź wykres.

| | A | B | C | D | E | F | G | H |
|---|---|---|---|---|---|---|---|---|
| 1 | **wyniki wyścigów – liga kobieca** | | | | **dystans [km]** | 10,00 | | |
| 2 | **miejsce** | **zawodniczka** | **czas [mm:ss]** | **strata do najlepszego [mm:ss]** | **minuty** | **sekundy** | **na godziny** | **prędkość [km/h]** |
| 3 | 1 | Agata | 17:21 | | | | | |
| 4 | 2 | Zosia | 17:27 | | | | | |
| 5 | 3 | Malwina | 17:29 | | | | | |
| 6 | 4 | Ilona | 17:35 | | | | | |
| 7 | 5 | Agnieszka | 17:39 | | | | | |
| 8 | 6 | Magda | 17:52 | | | | | |
| 9 | 7 | Kasia W. | 18:23 | | | | | |
| 10 | 8 | Kasia J. | 18:24 | | | | | |
| 11 | 9 | Aga | 18:26 | | | | | |
| 12 | 9 | Izabela | 18:26 | | | | | |

## PODSUMOWANIE

- Arkusz kalkulacyjny to wygodne narzędzie analityczne, które umożliwia mniej i bardziej skomplikowane obliczenia oraz graficzną prezentację danych. Na co dzień przydaje się do monitorowania postępów w realizacji założonego celu, pozwala planować i kontrolować budżet czy proces nauki.
- Podczas sporządzania zestawienia trzeba dbać o odpowiedni format liczb oraz czytelność tabeli.
- Do przeprowadzenia obliczeń można wykorzystywać funkcje wbudowane w arkusz, takie jak data i godzina, matematyczne, trygonometryczne, statystyczne.
- W formułach należy stosować adresy względne, bezwzględne lub mieszane – w zależności od tego, w jaki sposób arkusz ma zmieniać adresy podczas kopiowania formuł między komórkami.
- Odpowiednio opracowane dane, ujęte w formie tabelarycznej, można przedstawić w postaci graficznej. Często ułatwia to zrozumienie informacji oraz zależności między nimi. Należy dobrać typ wykresu w zależności od konkretnych danych: wykresy słupkowy i kolumnowy służą do porównywania wartości liczbowych, kołowy i pierścieniowy – do analizy procentowego udziału danej kategorii, a liniowy – do badania danych zmieniających się w czasie.

## PYTANIA SPRAWDZAJĄCE

1. Czy format liczby ma wpływ na wartość komórki? Uzasadnij.
2. Kiedy stosuje się adresy względne, a kiedy bezwzględne i mieszane?
3. Na jakim wykresie przedstawisz wyniki zawodów sportowych, gdy masz podane dla trzech rund: miejsce oraz liczbę zdobytych punktów? Uzasadnij swój wybór.

## PYTANIA QUIZOWE

1. Która z podanych formuł sumowania bloku komórek jest poprawna?
   a. `SUMA(A1:B10)`
   b. `=SUMA[A1:B10]`
   c. `=SUMA(A1:B10)`
   d. `=SUMA(A1-B10)`
2. Wpisano formułę `=B2/$B$7*100%`. Jak zmienią się odwołania przy kopiowaniu (przeciąganiu) formuły w dół?
   a. Odwołania do komórek się nie zmienią.
   b. Pierwsze odwołanie się zmieni, drugie – nie.
   c. Pierwsze odwołanie się nie zmieni, drugie – tak.
   d. Oba odwołania się zmienią.

3. Co oznacza zapis `=MIN(D:D)`?
   **a.** Formuła zwróci wartość minimalną w kolumnie **D**.
   **b.** Formuła zwróci wartość maksymalną w wierszu **D**.
   **c.** Formuła zwróci wartość minimalną w wierszu **D**.
   **d.** Formuła jest błędna, nie można jej poprawnie zinterpretować.

## ZADANIA DODATKOWE

### Zadanie 1. Wyniki zawodów

Wyszukaj w internecie wyniki zawodów rozgrywanych w twojej okolicy, np. pływackich albo lekkoatletycznych, a następnie sporządź w arkuszu kalkulacyjnym ranking i przedstaw dane na wykresie.

### Zadanie 2. Kalkulator rowerowy

Zapoznaj się z kalkulatorem rowerowym na stronie **www.bikecalc.com** i przygotuj wybrane zestawienie w arkuszu kalkulacyjnym.

Wheel — Chainring — Cog — Cadence: 90

Rim Size: 700c/29er (622mm) — min: 40 — 10 — Even toothed gears only?

Tire Size: 23 — max: 48 — 20 — Ambidextrous skidder?

Target Speed: 18 mph: ◉ kph: ○

Calculate

*What gear will give me 18 mph?*

| | *Slower* | | *Closest* | | *Faster* |
|---|---|---|---|---|---|
| *Speed* | 17.82 | 17.82 | **18.03** | 18.03 | 18.24 |
| *Gear* | 48x19 | 43x17 | **46x18** | 41x16 | 44x17 |
| *Skid Patches* | 19 | 17 | **9** | 16 | 17 |

*How fast will I be going at various cadences using these gears?*

| *Cad / Gear* | *30* | *40* | *50* | *60* | *70* | *80* | *90* | *100* | *110* | *120* | *130* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| *48x19* | 5.94 | 7.92 | 9.9 | 11.88 | 13.86 | 15.84 | 17.82 | 19.79 | 21.77 | 23.75 | 25.73 |
| *43x17* | 5.94 | 7.92 | 9.9 | 11.88 | 13.86 | 15.84 | 17.82 | 19.79 | 21.77 | 23.75 | 25.73 |
| *46x18* | 6.01 | 8.01 | 10.01 | 12.02 | 14.02 | 16.02 | **18.03** | 20.03 | 22.03 | 24.04 | 26.04 |
| *41x16* | 6.01 | 8.01 | 10.01 | 12.02 | 14.02 | 16.02 | 18.03 | 20.03 | 22.03 | 24.04 | 26.04 |
| *44x17* | 6.08 | 8.11 | 10.13 | 12.16 | 14.18 | 16.21 | 18.24 | 20.26 | 22.29 | 24.32 | 26.34 |

### Zadanie 3. Trochę historii

Pomysł arkusza kalkulacyjnego zrodził się ponad czterdzieści lat temu. Żmudne obliczenia, które trzeba było wykonywać na papierze, stały się inspiracją dla Dana Bricklina (wtedy studenta Harvard Business School) do stworzenia prototypu arkusza kalkulacyjnego, który ponoć miał wpływ na rozpowszechnienie się komputerów osobistych. Poszukaj informacji na ten temat. Sprawdź, kto i kiedy opracował pierwsze arkusze.



# 3. Stosowanie instrukcji warunkowych

- Rodzaje formuł warunkowych
- Składnia instrukcji warunkowych
- Praktyczne wykorzystanie formuł warunkowych

## ZADANIE NA START

**Ile osób może mówić prawdę?**

Ania, Michał i Janek zdawali egzamin z języka obcego. Teraz relacjonują wyniki.

**Ania:** Michał i ja zdaliśmy egzamin.

**Michał:** Janek i ja zdaliśmy egzamin.

**Janek:** Ania i ja zdaliśmy egzamin.

Ile osób może mówić prawdę, a ile kłamać? Przeanalizuj powyższy problem. Przyjmij, że dana wypowiedź ma wartość „prawda" albo „fałsz", i rozpatrz kolejne warianty, np.

- Jeżeli Ania, Michał i Janek mówią prawdę, to…
- Jeżeli Ania, Michał i Janek kłamią, to…




## INSTRUKCJA WARUNKOWA

Instrukcja warunkowa jest konstrukcją programistyczną, która określa dalsze działanie w zależności od tego, czy dane wyrażenie jest prawdziwe, czy fałszywe. W arkuszu kalkulacyjnym przyjmuje postać wbudowanych formuł, m.in. **JEŻELI**, **LICZ.JEŻELI**, **SUMA.JEŻELI** i **WYSZUKAJ.PIONOWO**.

**POSZUKAJ W INTERNECIE**

Dowiedz się, do czego można wykorzystać formułę warunkową **JEŻELI.BŁĄD**.

## FUNKCJA JEŻELI

Funkcja logiczna **JEŻELI** sprawdza, czy spełniony jest **warunek** (np. **C2>B2**). Jeżeli warunek jest spełniony, to wartością funkcji jest **wartość1** (np. "**PRAWDA**"). Jeżeli warunek nie jest spełniony, wynikiem funkcji jest **wartość2** (np. "**FAŁSZ**").

**ZAPAMIĘTAJ!**

`=JEŻELI(warunek;wartość1;wartość2)`

## Ćwiczenie 1. Badanie realizacji planu nauki

Przygotuj tabelę, za pomocą której zbadasz realizację planu nauki słów w języku obcym (wzór poniżej). Załóż, że każdego dnia nauczysz się co najmniej 20 słów. Uzupełnij kolumnę **Realizacja planu**.

| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | **Dzień** | **Liczba słów** | **Realizacja planu** | | **Plan dzienny** | **20** |
| 2 | 1 | 32 | plan zrealizowany | | | |
| 3 | 2 | 19 | nie udało się | | | |
| 4 | 3 | 12 | nie udało się | | | |
| 5 | 4 | 25 | plan zrealizowany | | | |
| 6 | 5 | 17 | nie udało się | | | |
| 7 | 6 | 20 | plan zrealizowany | | | |

1. Sformułuj warunek opisowy.
   Jeżeli liczba słów jest większa niż założony plan dzienny (20 słów) lub równa temu planowi, napisz „plan zrealizowany”, w przeciwnym razie napisz „nie udało się”.
2. Zapisz odpowiednią formułę w komórce **C2**. Komórka z liczbą słów powinna mieć adres względny **B2**, a komórka z planem dziennym – adres bezwzględny **$F$1**, aby przy kopiowaniu formuły pierwszy adres się zmieniał, a drugi nie.
   `=JEŻELI(B2>=$F$1;"plan zrealizowany";"nie udało się")`
3. Skopiuj formułę z komórki **C2** do pozostałych komórek.

| | A | B | C |
|---|---|---|---|
| 1 | **Dzień** | **Liczba słów** | **Realizacja planu** |
| 2 | 1 | 32 | =JEŻELI(B2>=$F$1;"plan zrealizowany";"nie udało się") |
| 3 | 2 | 19 | =JEŻELI(B3>=$F$1;"plan zrealizowany";"nie udało się") |
| 4 | 3 | 12 | =JEŻELI(B4>=$F$1;"plan zrealizowany";"nie udało się") |

**W PRAKTYCE**

Z arkuszy Google i Excela można też korzystać na urządzeniach mobilnych – przez przeglądarkę lub aplikację przeznaczoną do tego celu. W przypadku Excela należy pamiętać, że w wersjach web i mobilnej nie obsługuje on wszystkich formatów plików, niektóre funkcje nie są aktywne lub działają inaczej niż w programie desktopowym. Aplikacja Google również ma nieco bardziej ograniczone możliwości.

W przypadku bardziej skomplikowanych zestawień i obliczeń stosuje się funkcje zagnieżdżone, tzn. wpisuje się w miejsce np. warunku lub wartości inną funkcję. Dzięki temu nie trzeba tworzyć dodatkowych kolumn prowadzących do wyniku końcowego albo – jak w przypadku zagnieżdżenia kilku funkcji **JEŻELI** – można sprawdzać

jednocześnie wiele kryteriów. W trakcie tworzenia tego typu formuł należy pamiętać o zamykaniu nawiasów funkcji – inaczej wystąpi błąd.

## Ćwiczenie 2. Planowanie powtórzenia

Przygotuj tabelę, za pomocą której zaplanujesz powtórzenie do egzaminu z języka obcego: czytanie ze zrozumieniem co 3 dni, gramatyka co 2 dni, pisanie wypracowań co 5 dni (wzór poniżej). Uzupełnij kolumny **Czytanie ze zrozumieniem**, **Gramatyka** i **Pisanie wypracowań**.

| | A | B | C | D |
|---|---|---|---|---|
| 1 | **Typ zadania** | **Czytanie ze zrozumieniem** | **Gramatyka** | **Pisanie wypracowań** |
| 2 | **Co ile dni** | **3** | **2** | **5** |
| 3 | 1 | | | |
| 4 | 2 | | TAK | |
| 5 | 3 | TAK | | |
| 6 | 4 | | TAK | |
| 7 | 5 | | | TAK |
| 8 | 6 | TAK | TAK | |

1. Sformułuj warunek opisowy.
   Aby stwierdzić, jakiego typu zadania mają być realizowane danego dnia, podziel liczbę oznaczającą numer dnia przez liczbę określającą częstość realizacji danego typu zadań i ustal wartość reszty z dzielenia: jeżeli reszta wynosi 0, wpisz komunikat „TAK”, w przeciwnym razie pozostaw pustą komórkę.
2. Zapisz odpowiednią formułę w komórce **B3**.
   - Aby wyznaczyć resztę z dzielenia **liczba1** przez **liczba2**, skorzystaj z funkcji matematycznej **MOD(liczba1;liczba2)**. Wyrażenie może mieć wartość dodatnią lub 0.
   - Aby jedna formuła realizowała wyliczenie dla całego zestawu, komórki z liczbami oznaczającymi numer dnia i częstość realizacji danego typu zadań powinny mieć adres mieszany. Dzięki temu podczas kopiowania formuły będą odwoływać się do tej samej kolumny (**A**) i do tego samego wiersza (**2**).
     `=JEŻELI(MOD($A3;B$2)=0;"TAK";"")`
3. Skopiuj formułę z komórki **B3** do pozostałych komórek – przeciągnij ją w dół i w prawo.

| | B | C | D |
|---|---|---|---|
| 1 | **Czytanie ze zrozumieniem** | **Gramatyka** | **Pisanie wypracowań** |
| 2 | **3** | **2** | **5** |
| 3 | =JEŻELI(MOD($A3;B$2)=0;"TAK";"") | =JEŻELI(MOD($A3;C$2)=0;"TAK";"") | =JEŻELI(MOD($A3;D$2)=0;"TAK";"") |
| 4 | =JEŻELI(MOD($A4;B$2)=0;"TAK";"") | =JEŻELI(MOD($A4;C$2)=0;"TAK";"") | =JEŻELI(MOD($A4;D$2)=0;"TAK";"") |
| 5 | =JEŻELI(MOD($A5;B$2)=0;"TAK";"") | =JEŻELI(MOD($A5;C$2)=0;"TAK";"") | =JEŻELI(MOD($A5;D$2)=0;"TAK";"") |
| 6 | =JEŻELI(MOD($A6;B$2)=0;"TAK";"") | =JEŻELI(MOD($A6;C$2)=0;"TAK";"") | =JEŻELI(MOD($A6;D$2)=0;"TAK";"") |

## FUNKCJA LICZ.JEŻELI

Funkcja statystyczna **LICZ.JEŻELI** umożliwia zliczenie tych komórek z danego obszaru, które spełniają dane kryterium. Przyjmuje dwa argumenty: **zakres** (przeszukiwane dane, np. **A1:A25** albo **A:A** w przypadku całej kolumny) oraz **kryterium** (określony warunek, np. **">100"**).

**ZAPAMIĘTAJ!**

```
=LICZ.JEŻELI(zakres;kryterium)
```

Aby sprawdzić, ile wymienionych w tabeli osób:

- zdobyło najwyżej 20 punktów, należy zapisać kryterium w postaci **"<=20"**;
- zaliczyło test w pierwszym kwartale 2018 roku, należy zapisać kryterium w postaci **"<=2018-03-31"**;
- otrzymało ocenę różną od ndst., należy zapisać kryterium w postaci **"<>ndst."**.

## Ćwiczenie 3. Odnotowywanie postępów w nauce

Przygotuj tabelę, za pomocą której odnotujesz wyniki testów próbnych i postępy w nauce (wzór poniżej). Zastosuj odpowiednie funkcje warunkowe, aby uzupełnić:

a. kolumnę **Ocena** zawierającą ocenę postępów w nauce w odniesieniu do poprzedniego dnia,
b. kolumnę **Ile razy** zliczającą, ile razy było lepiej, gorzej, bez zmian.

| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | **Dzień** | **Wynik testu** | **Ocena** | | **Postępy** | **Ile razy** |
| 2 | 1 | 120 | | | lepiej | 3 |
| 3 | 2 | 123 | lepiej | | gorzej | 4 |
| 4 | 3 | 101 | gorzej | | bez zmian | 1 |
| 5 | 4 | 98 | gorzej | | | |
| 6 | 5 | 117 | lepiej | | | |
| 7 | 6 | 117 | bez zmian | | | |
| 8 | 7 | 109 | gorzej | | | |
| 9 | 8 | 80 | gorzej | | | |
| 10 | 9 | 95 | lepiej | | | |

1. Sformułuj warunek opisowy dotyczący kolumny **Ocena**.
   Jeżeli wartość w danej komórce w kolumnie **B** jest większa niż wartość w komórce poprzedniej, napisz „lepiej”, w przeciwnym razie sprawdź drugi warunek: jeżeli wartość w danej komórce jest mniejsza niż wartość w komórce poprzedniej, napisz „gorzej”; jeśli obie wartości są równe, napisz „bez zmian”.
2. Zapisz odpowiednią formułę w komórce **C3**.

```
=JEŻELI(B3>B2;"lepiej";JEŻELI(B3<B2;"gorzej";"bez zmian"))
```

3. Aby wykonać polecenie z punktu **a**, skopiuj formułę z komórki **C3** do pozostałych komórek w tej kolumnie.

| | A | B | C |
|---|---|---|---|
| 1 | **Dzień** | **Wynik testu** | **Ocena** |
| 2 | 1 | 120 | |
| 3 | 2 | 123 | =JEŻELI(B3>B2;"lepiej";JEŻELI(B3<B2;"gorzej";"bez zmian")) |
| 4 | 3 | 101 | =JEŻELI(B4>B3;"lepiej";JEŻELI(B4<B3;"gorzej";"bez zmian")) |
| 5 | 4 | 98 | =JEŻELI(B5>B4;"lepiej";JEŻELI(B5<B4;"gorzej";"bez zmian")) |

4. Sformułuj warunek opisowy dotyczący kolumny **Ile razy**.
Jeżeli w kolumnie **Ocena** znajdują się komórki spełniające dane kryterium ("lepiej", "gorzej", "bez zmian"), oblicz liczbę takich komórek.

5. Wprowadź odpowiednią formułę w komórce **F2.**

```
=LICZ.JEŻELI(C:C;E2)
```

6. Aby wykonać polecenie z punktu **b**, skopiuj formułę z komórki **F2** do pozostałych komórek w tej kolumnie.

| | E | F |
|---|---|---|
| 1 | **Postępy** | **Ile razy** |
| 2 | lepiej | =LICZ.JEŻELI(C:C;E2) |
| 3 | gorzej | =LICZ.JEŻELI(C:C;E3) |
| 4 | bez zmian | =LICZ.JEŻELI(C:C;E4) |

## FUNKCJA SUMA.JEŻELI

Funkcja matematyczna **SUMA.JEŻELI** umożliwia zbadanie, czy komórki z danego zakresu spełniają kryterium, i zsumowanie na tej podstawie wartości należących do innego zakresu. Ta funkcja przyjmuje trzy argumenty: **zakres1** (przeszukiwane dane, np. **B:B**), **kryterium** (określony warunek) oraz **zakres2** (zakres danych do zsumowania, odpowiadający argumentowi **zakres1**), przy czym **zakres2** to argument opcjonalny – jeśli zostanie pominięty, program zsumuje komórki wyznaczone przez argument **zakres1**. Aby np. zsumować w kolumnie **A** wszystkie liczby większe od 10, należy zastosować formułę **=SUMA.JEŻELI(A:A;>10)**.

**ZAPAMIĘTAJ!**

```
=SUMA.JEŻELI(zakres1;kryterium;zakres2)
```

## Ćwiczenie 4. Zestawianie wyników egzaminu próbnego

Przygotuj tabelę, za pomocą której zestawisz wyniki egzaminu (wzór na stronie obok). Zastosuj odpowiednie formuły, aby uzupełnić:

a. kolumnę **Suma** prezentującą łączną liczbę punktów zdobytych podczas egzaminu,

**b.** kolumnę **%** prezentującą wynik procentowy egzaminu,
**c.** kolumnę **Zal/Nzal** prezentującą informację, czy egzamin został zaliczony, czy też nie,
**d.** średnią liczbę punktów dla egzaminów zaliczonych i niezaliczonych.

| | A | B | C | D | E | F | G | H | I | J | K | L | M |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | **Egzamin** | **Osoba** | **Czytanie** | **Pisanie** | **Słuchanie** | **Mówienie** | **Gramatyka** | **Suma** | **%** | **Zal/Nzal** | | **egzamin zaliczony:** | |
| 2 | | | **40** | **40** | **40** | **40** | **40** | **200** | **100%** | | | wynik ≥ 60% | |
| 3 | | Basia | 12 | 32 | 15 | 16 | 13 | 88 | 44% | nzal | | | |
| 4 | I | Janek | 20 | 22 | 20 | 17 | 20 | 99 | 50% | nzal | | suma, gdy zal | 796 |
| 5 | | Witek | 15 | 25 | 35 | 35 | 35 | 145 | 73% | zal | | suma, gdy nzal | 414 |
| 6 | | Basia | 14 | 28 | 35 | 32 | 12 | 121 | 61% | zal | | | |
| 7 | II | Janek | 24 | 23 | 24 | 21 | 26 | 118 | 59% | nzal | | ile zal | 5 |
| 8 | | Witek | 12 | 21 | 32 | 23 | 21 | 109 | 55% | nzal | | ile nzal | 4 |
| 9 | | Basia | 38 | 32 | 32 | 33 | 37 | 172 | 86% | zal | | | |
| 10 | III | Janek | 35 | 37 | 38 | 31 | 40 | 181 | 91% | zal | | średnia zal | 159,20 |
| 11 | | Witek | 29 | 39 | 35 | 39 | 35 | 177 | 89% | zal | | średnia nzal | 103,50 |

1. Wyznacz łączną liczbę uzyskanych punktów w komórce **H3**.
   `=SUMA(C3:G3)`

2. Aby wykonać polecenie z punktu **a**, skopiuj formułę z komórki **H3** do pozostałych komórek w tej kolumnie.

3. Wyznacz w komórce **I3**, ile procent stanowią uzyskane punkty. Pamiętaj o odpowiednim formacie liczb.
   `=H3/$H$2`

4. Aby wykonać polecenie z punktu **b**, skopiuj formułę z komórki **I3** do pozostałych komórek w tej kolumnie. (Jeśli po skopiowaniu formuły w niektórych komórkach uzyskasz ułamek dziesiętny, ustaw wyświetlanie wartości z zaokrągleniem do części całkowitych).

5. Sformułuj warunek opisowy dotyczący kolumny **Zal/Nzal** i wprowadź odpowiednią formułę w komórce **J3.**
   `=JEŻELI(I3>=60%;"zal";"nzal")`

6. Aby wykonać polecenie z punktu **c**, skopiuj formułę z komórki **J3** do pozostałych komórek w tej kolumnie.

| | A | B | H | I | J |
|---|---|---|---|---|---|
| 1 | **Egzamin** | **Osoba** | **Suma** | **%** | **Zal/Nzal** |
| 2 | | | =SUMA(C2:G2) | =H2/200 | |
| 3 | | Basia | =SUMA(C3:G3) | =H3/$H$2 | =JEŻELI(I3>60%;"zal";"nzal") |
| 4 | I | Janek | =SUMA(C4:G4) | =H4/$H$2 | =JEŻELI(I4>60%;"zal";"nzal") |
| 5 | | Witek | =SUMA(C5:G5) | =H5/$H$2 | =JEŻELI(I5>60%;"zal";"nzal") |

7. Wyznacz średnią liczbę uzyskanych punktów dla egzaminów zaliczonych i niezaliczonych.

- Oblicz łączną liczbę uzyskanych punktów za pomocą funkcji **SUMA.JEŻELI**.
  `=SUMA.JEŻELI(J:J;"zal";H:H)`
  `=SUMA.JEŻELI(J:J;"nzal";H:H)`

- Oblicz liczbę zaliczonych i niezaliczonych egzaminów za pomocą funkcji **LICZ.JEŻELI**.

  `=LICZ.JEŻELI(J:J;"zal")`

  `=LICZ.JEŻELI(J:J;"nzal")`

- Aby wykonać polecenie z punktu **d**, podziel wyniki funkcji **SUMA.JEŻELI** przez wyniki funkcji **LICZ.JEŻELI**.

| | L | M |
|---|---|---|
| 4 | suma, gdy zal | =SUMA.JEŻELI(J:J;"zal";H:H) |
| 5 | suma, gdy nzal | =SUMA.JEŻELI(J:J;"nzal";H:H) |
| 6 | | |
| 7 | ile zal | =LICZ.JEŻELI(J:J;"zal") |
| 8 | ile nzal | =LICZ.JEŻELI(J:J;"nzal") |
| 9 | | |
| 10 | średnia zal | =M4/M7 |
| 11 | średnia nzal | =M5/M8 |

### Ćwiczenie 5. Opracowanie arkusza do systematycznej pracy

Przygotuj i uzupełnij arkusz złożony co najmniej z trzech skoroszytów, który będzie wspierać cię w systematycznej realizacji jakiegoś zadania, np. w przygotowaniach do zawodów sportowych. Wykorzystaj funkcje **JEŻELI** w wersjach prostej i zagnieżdżonej, **LICZ.JEŻELI**, **SUMA.JEŻELI**.

## PODSUMOWANIE

- Instrukcje warunkowe systematyzują analizę danych – umożliwiają sprawdzenie, czy określone warunki są spełnione, i wyliczenie na tej podstawie wartości wyrażeń.
- Funkcja **JEŻELI** sprawdza, czy dany warunek jest spełniony, i wykonuje odpowiednią operację. Jej składnia ma postać **=JEŻELI(warunek;wartość1;wartość2)**.
- Funkcja **LICZ.JEŻELI** umożliwia wyznaczenie liczby komórek, które spełniają dane kryteria. Jej składnia ma postać **=LICZ.JEŻELI(zakres;kryterium)**.
- Funkcja **SUMA.JEŻELI** sumuje zakres danych spełniających określone kryteria. Jej składnia ma postać **=SUMA.JEŻELI(zakres1;kryterium;zakres2)**.

## PYTANIA SPRAWDZAJĄCE

1. Jakie znasz funkcje warunkowe? Do czego służą?
2. Jakiego typu wartości są wynikami poznanych funkcji warunkowych?
3. Jakie jest zastosowanie funkcji **LICZ.JEŻELI**? Podaj przykłady.

## PYTANIA QUIZOWE

1. Która z poniższych liczb stanowi wynik funkcji **`=MOD(99;2)`**?
   **a.** 0
   **b.** 1
   **c.** 2
   **d.** 3
2. Jakie wartości powinny być wpisane w komórki **`B2`** i **`B3`**, aby wynikiem formuły **`=JEŻELI(B3>B2;"lepiej";JEŻELI(B3<B2;"gorzej";"bez zmian"))`** był komunikat **`"bez zmian"`**?
   **a.** Odpowiednio 11 i 13.
   **b.** Odpowiednio 13 i 11.
   **c.** Odpowiednio 11 i 11.
   **d.** Odpowiednio 0 i 11.
3. Egzamin jest zaliczany, jeśli uczeń uzyska co najmniej 65 %. Jaką formułę należy wpisać w arkuszu w komórkę **`B3`**, aby uzyskać poprawny wynik?
   **a.** **`=JEŻELI(B3>=65%;"nzal";"zal")`**
   **b.** **`=JEŻELI(B3>=65%;"zal";"nzal")`**
   **c.** **`=JEŻELI(B3>65%;"zal";"nzal")`**
   **d.** **`=JEŻELI(B3>65%;"nzal";"zal")`**

## ZADANIA DODATKOWE

### Zadanie 1. Funkcja WYSZUKAJ.PIONOWO

Przygotuj tabelę, za pomocą której zestawisz liczbę punktów z egzaminu końcowego (wzór poniżej). Wyznacz łączną liczbę punktów uzyskanych przez każdego ucznia oraz przyporządkuj każdej łącznej liczbie punktów ocenę według podanej skali.

| | A | B | C | D | E | F | G | H | I | J | K |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | **Uczeń** | **Czytanie** | **Pisanie** | **Słuchanie** | **Mówienie** | **Gramatyka** | **Razem** | **Ocena** | | | |
| 2 | | **40** | **40** | **40** | **40** | **40** | | | | **Skala** | |
| 3 | uczeń 1 | 33 | 28 | 29 | 31 | 39 | 160 | db | | 0 | ndst |
| 4 | uczeń 2 | 27 | 31 | 25 | 30 | 27 | 140 | dst | | 120 | dop |
| 5 | uczeń 3 | 30 | 37 | 33 | 31 | 31 | 162 | db | | 140 | dst |
| 6 | uczeń 4 | 40 | 32 | 38 | 31 | 39 | 180 | bdb | | 160 | db |
| 7 | uczeń 5 | 36 | 12 | 28 | 32 | 29 | 137 | dop | | 180 | bdb |
| 8 | uczeń 6 | 29 | 33 | 39 | 30 | 33 | 164 | db | | 195 | cel |
| 9 | uczeń 7 | 31 | 36 | 30 | 12 | 26 | 135 | dop | | | |
| 10 | uczeń 8 | 38 | 29 | 27 | 28 | 38 | 160 | db | | | |
| 11 | uczeń 9 | 26 | 38 | 36 | 40 | 28 | 168 | db | | | |

**Wskazówka**: Wykorzystaj funkcję wyszukiwania i odwołań (adresu) **`WYSZUKAJ.PIONOWO`**, która umożliwia znajdowanie danych w podanym zakresie według wierszy. Funkcja przyjmuje trzy argumenty: **`wartość`** (szukaną wartość, np. liczbę uzyskanych punktów), **`zakres`** (przeszukiwane dane) oraz **`indeks`** (numer kolumny w zakresie zawierającym wartość będącą wynikiem funkcji).

### Zadanie 2. Sudoku – wstęp

Sudoku to gra logiczna polegająca na uzupełnieniu pustych pól tabeli w taki sposób, aby w każdym wierszu, w każdej kolumnie i w każdym dziewięciopolowym kwadracie 3 × 3 znalazły się liczby od 1 do 9. Przygotuj formuły, które sprawdzą poprawność w pionie i w poziomie dla trzech poniższych kwadratów.

| | | |
|---|---|---|
| 5 | 1 | 7 |
| 6 | 8 | 3 |
| 9 | 4 | 2 |

| | | |
|---|---|---|
| 5 | 1 | 7 |
| 6 | 8 | 8 |
| 9 | 4 | 2 |

| | | |
|---|---|---|
| 5 | 1 | 7 |
| 1 | 8 | 3 |
| 9 | 4 | 2 |

### Zadanie 3. Sudoku – gra

Uzupełnij puste pola poniższego kwadratu, a następnie wstaw formuły sprawdzające poprawność rozwiązania.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 5 | | | | 6 | | 3 | | |
| | | | | | | 9 | 1 | |
| | | 2 | | 3 | | | | |
| | | | | | | | 6 | |
| | | | 5 | 4 | | | | 9 |
| 4 | | 1 | 2 | | | 5 | 7 | |
| | 9 | | | | 7 | | | 1 |
| | | | 4 | | | | | |
| | | | | 5 | 3 | 6 | 4 | |

**Wskazówka:** pamiętaj także o sprawdzaniu w kwadracie 3 × 3.



# 4. Porównywanie ofert w arkuszu

- Opracowanie modelu
- Posługiwanie się instrukcją warunkową
- Projektowanie interfejsu użytkownika

## ZADANIE NA START

### Którą ofertę wybrać

Oferta aeroklubu zawiera cztery opcje w zależności od wybranego pakietu. Którą ofertę należy wybrać, aby zminimalizować koszty wynajmu szybowca, gdy pilot zamierza wylatać 45 godzin, a którą, gdy chce wylatać 72 godziny?



Aeroklub **Jaskółka**

CENNIK

| | BEZ PAKIETU | PAKIET 30 | PAKIET 50 | PAKIET 80 |
|---|---|---|---|---|
| Liczba godzin lotu w pakiecie | 0 | 30 | 50 | 80 |
| Cena pakietu [zł] | 0 | 1260 | 2000 | 3000 |
| Cena za 1 h lotu poza pakietem [zł] | 46 | | | |

## OPRACOWANIE MODELU

Warto porównywać oferty. Często firmy reklamujące swoje usługi eksponują mocne strony oferty, które mogą przyciągnąć klientów. Dopiero rzeczowa analiza – traktowana jako problem algorytmiczny – pozwala na wybór najkorzystniejszej opcji.

W trakcie rozwiązywania problemu algorytmicznego z wykorzystaniem możliwości arkusza kalkulacyjnego tworzy się pewien model – problem zapisuje się w sposób bardziej formalny, następnie wybiera się algorytm rozwiązania i przechodzi do jego komputerowej realizacji, która jest zakończona testowaniem.

Przy opracowywaniu w arkuszu symulatora umożliwiającego porównanie ofert warto postępować według poniższego planu.

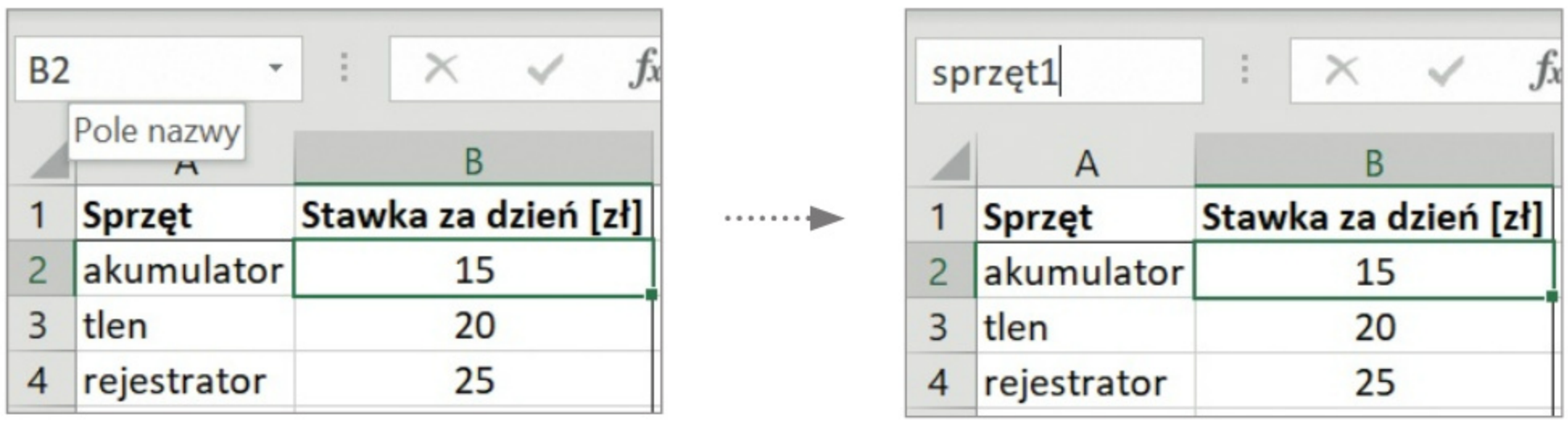
1. Wprowadzenie nagłówków oraz danych liczbowych.
2. Wprowadzenie nazw komórek i zakresów komórek.
3. Wprowadzenie formuł, które pozwolą wykonać obliczenia.
4. Wprowadzenie elementów ułatwiających analizę, np. paska przewijania do szybkiej zmiany liczby godzin wylatanych przez pilota.

Należy pamiętać, że arkusz kalkulacyjny jest narzędziem, które wykonuje obliczenia, jednak to użytkownik wprowadza dane i algorytm w postaci formuł oraz ostatecznie podejmuje decyzję o wyborze oferty. W dodatku zwykle podczas budowania modelu stosuje się pewne uproszczenia, pomija się niektóre wartości albo zależności. Z jednej strony pozwala to szybko otrzymać wynik, lecz z drugiej – w niektórych przypadkach uproszczenia mogą negatywnie wpłynąć na ten wynik.

## NAZWY KOMÓREK I ZAKRESÓW KOMÓREK

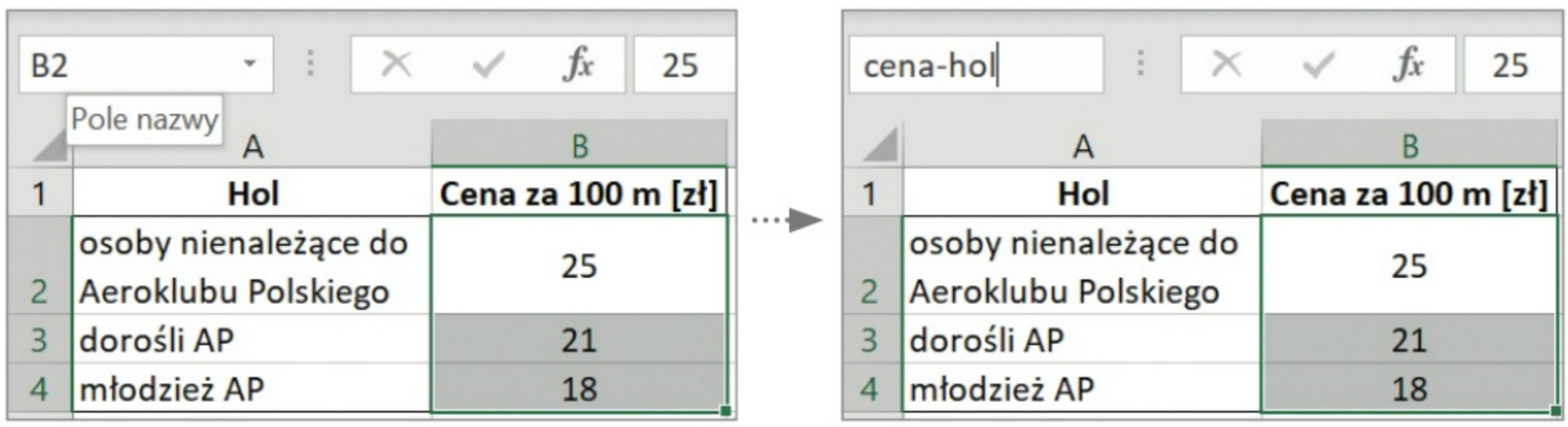
Dzięki zastosowaniu nazw komórek i zakresów komórek formuły zapisywane w arkuszu stają się bardziej zrozumiałe i łatwiejsze do opracowania.

Najprostszy sposób definiowania nazwy komórki to zaznaczenie tej komórki (np. **B2**) i wpisanie odpowiedniej treści (np. **sprzęt1**) w pole nazwy na pasku formuły.




**Rys. 1.** Nadawanie nazwy komórce **B2** w programie Excel

Nazwy zakresów komórek definiuje się analogicznie – wystarczy zaznaczyć odpowiednie komórki (np. **B2:B4**) i wpisać odpowiednią treść (np. **cena-hol**) w pole nazwy na pasku formuły.




**Rys. 2.** Nadawanie nazwy zakresowi komórek **B2:B4** w programie Excel

## Ćwiczenie 1. Opracowanie arkusza

Przygotuj w arkuszu kalkulacyjnym tabelę, która będzie stanowiła podstawę symulatora umożliwiającego wybór najkorzystniejszej opcji w zależności od liczby wylatanych godzin (wzór poniżej). Nadaj nazwy komórkom wykorzystywanym do obliczeń.

| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | | **Bez pakietu** | **Pakiet 30** | **Pakiet 50** | **Pakiet 80** |
| 2 | **Liczba godzin lotu w pakiecie** | 0 | 30 | 50 | 80 |
| 3 | **Cena pakietu [zł]** | 0 | 1260 | 2000 | 3000 |
| 4 | **Cena za 1 h lotu poza pakietem [zł]** | 46 | 46 | 46 | 46 |

1. Scal komórki **B4:E4** za pomocą opcji **Scal i wyśrodkuj**, zaznacz komórkę utworzoną w ten sposób i wpisz w pole nazwy na pasku formuły termin **`godzina_lotu`**.

2. Nadaj nazwy zakresom komórek zgodnie z rysunkiem poniżej – zaznacz zakresy, a potem wpisz treść w pole nazwy na pasku formuły.

| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | | **Bez pakietu** | **Pakiet 30** | **Pakiet 50** | **Pakiet 80** |
| 2 | **Liczba godzin lotu w pakiecie** | 0 | 30 | 50 | 80 |
| 3 | **Cena pakietu [zł]** | 0 | 1260 | 2000 | 3000 |
| 4 | **Cena za 1 h lotu poza pakietem [zł]** | 46 | | | |

godzina_lotu cena_pakietu godziny_w_pakiecie

3. Rozszerz arkusz o fragment, w którym będziesz wykonywać obliczenia. Nadaj nazwę komórce z liczbą wylatanych godzin oraz zakresowi, w którym będziesz obliczać wysokość opłat za latanie.

| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | | **Bez pakietu** | **Pakiet 30** | **Pakiet 50** | **Pakiet 80** |
| 2 | **Liczba godzin lotu w pakiecie** | 0 | 30 | 50 | 80 |
| 3 | **Cena pakietu [zł]** | 0 | 1260 | 2000 | 3000 |
| 4 | **Cena za 1 h lotu poza pakietem [zł]** | 46 | | | |
| 5 | | | | | |
| 6 | **Liczba wylatanych godzin** | | | | |
| 7 | **Opłata [zł]** | | | | |
| 8 | **Najkorzystniejsza oferta [zł]** | | | | |

ile

opłata

## Ćwiczenie 2. Analiza problemu i opracowanie formuł

Przeanalizuj tabelę utworzoną w ćwiczeniu 1, a następnie opracuj i wprowadź do niej formuły, za pomocą których arkusz obliczy opłatę dla kolejnych opcji oraz wskaże najkorzystniejszą ofertę.

1. Jeśli pilot lata bez pakietu, płaci za każdą godzinę lotu – opłata jest wprost proporcjonalna do długości lotu.

   **`=ile*godzina_lotu`**

2. Pakiet oznacza opłatę z góry za określoną liczbę godzin. Po przekroczeniu liczby godzin w pakiecie pilot płaci za godziny bez wykupionego pakietu.

   **`=cena_pakietu+(ile-godziny_w_pakiecie)*godzina_lotu`**

3. Jeśli liczba wylatanych godzin nie przekracza godzin w pakiecie, to cena jest stała, równa wartości pakietu. Jeśli liczba wylatanych godzin przekracza godziny w pakiecie, za każdą wylataną nadgodzinę pilot musi zapłacić.

```
=JEŻELI(ile<=godziny_w_pakiecie;cena_pakietu;
cena_pakietu+(ile-godziny_w_pakiecie)*godzina_lotu)
```

4. Aby arkusz wskazywał najkorzystniejszą ofertę, należy zastosować funkcje **JEŻELI** oraz **MIN** – docelowo w komórce ma się wyświetlić tylko wartość minimalna z zakresu komórek **opłata**, natomiast pozostałe komórki zastąpione zostaną spacją.

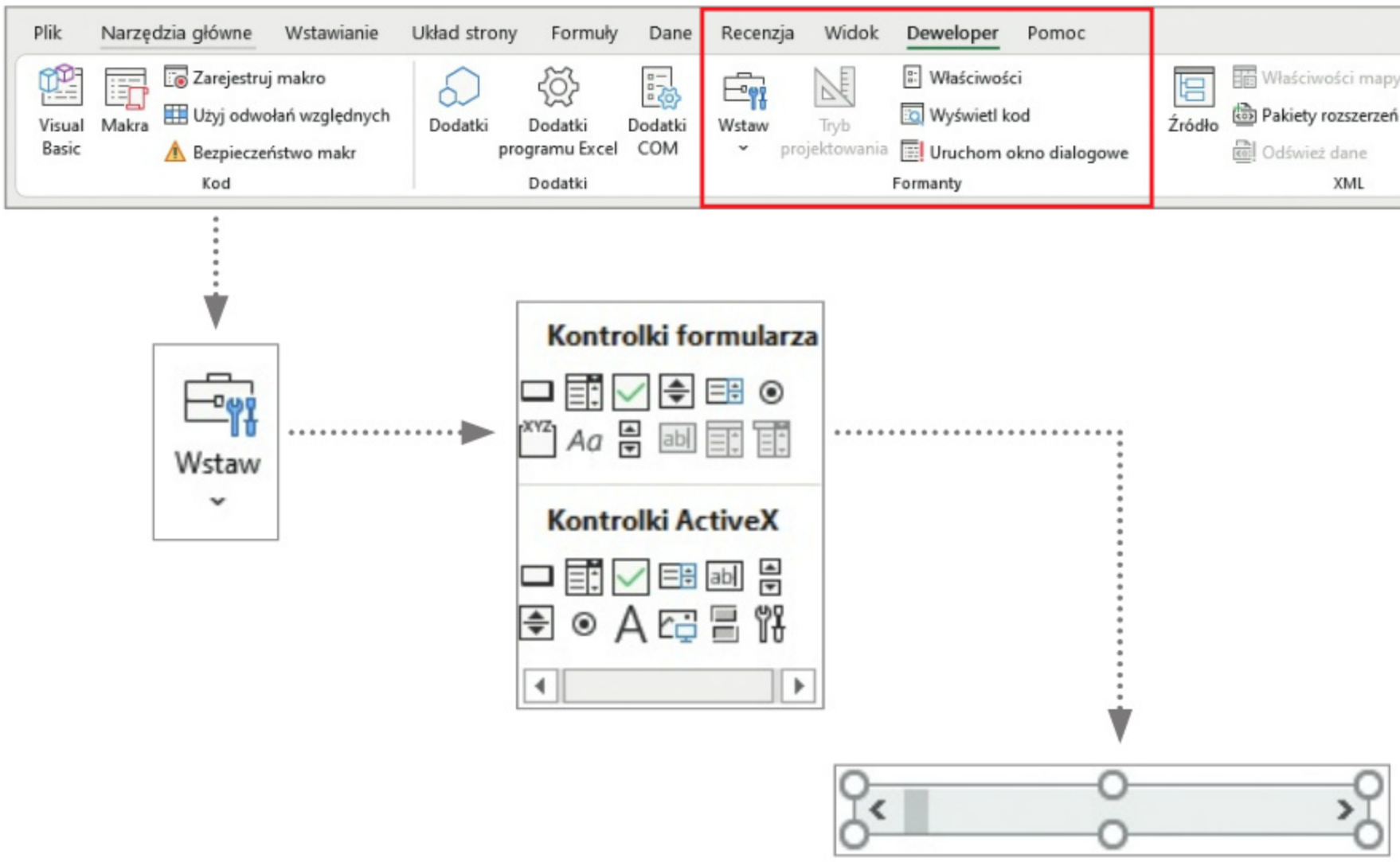
```
=JEŻELI(opłata=MIN(opłata);opłata;" ")
```

| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | | **Bez pakietu** | **Pakiet 30** | **Pakiet 50** | **Pakiet 80** |
| 2 | **Liczba godzin lotu w pakiecie** | 0 | 30 | 50 | 80 |
| 3 | **Cena pakietu [zł]** | 0 | 1260 | 2000 | 3000 |
| 4 | **Cena za 1 h lotu poza pakietem [zł]** | | 46 | | |
| 5 | | | | | |
| 6 | **Liczba wylatanych godzin** | 34 | | | |
| 7 | **Opłata [zł]** | 1564 | 1444 | 2000 | 3000 |
| 8 | **Najkorzystniejsza oferta [zł]** | | **1444** | | |

## INTERFEJS UŻYTKOWNIKA

Czytelnie przygotowany interfejs ułatwia szybką i bezbłędną zmianę danych wyjściowych. Przykładowo: pasek przewijania umożliwia zmianę danych wejściowych przez klikanie strzałek przewijania lub przeciąganie suwaka przewijania.

Aby wstawić pasek w Excelu, należy wybrać kolejno opcje **Deweloper → Wstaw → Kontrolki formularza → Pasek przewijania**, a następnie kliknąć arkusz w miejscu, w którym ma zostać umieszczony lewy górny narożnik paska (jeśli pasek ma mieć orientację pionową), albo przeciągnąć myszą w poziomie w obrębie dokumentu.




**Rys. 3.** Wstawianie paska przewijania w programie Excel

**W PRAKTYCE**

Jeśli karta **Deweloper** nie jest widoczna w arkuszu, należy wybrać **Plik** → **Opcje** → **Dostosowywanie Wstążki** i zaznaczyć ją wśród narzędzi z kart głównych.

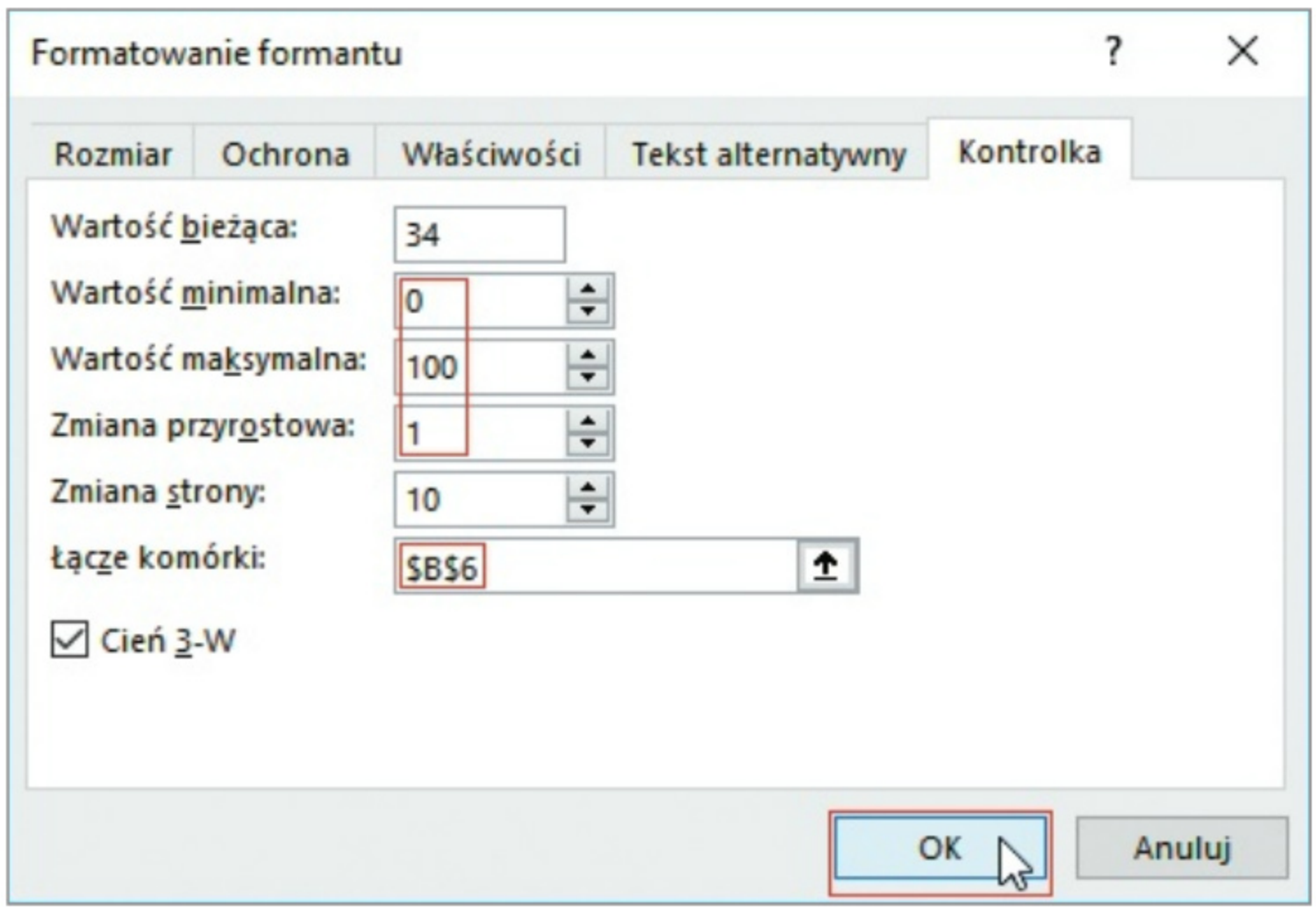
Aby zmienić orientację paska, trzeba przeciągnąć jeden z uchwytów zmiany rozmiaru po przekątnej. Po wybraniu w menu podręcznym opcji **Formatuj formant** można określić właściwości paska – wartości minimalną i maksymalną, zmianę przyrostową oraz łącze do komórki, w której ma się pojawiać wartość.

## Ćwiczenie 3. Projektowanie interfejsu użytkownika

Wstaw do symulatora z ćwiczenia 2 pasek, który umożliwi zmianę wartości liczby wylatanych godzin (wzór poniżej).

| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | | **Bez pakietu** | **Pakiet 30** | **Pakiet 50** | **Pakiet 80** |
| 2 | **Liczba godzin lotu w pakiecie** | 0 | 30 | 50 | 80 |
| 3 | **Cena pakietu [zł]** | 0 | 1260 | 2000 | 3000 |
| 4 | **Cena za 1 h lotu poza pakietem [zł]** | | 46 | | |
| 5 | | | | | |
| 6 | **Liczba wylatanych godzin** | 34 | < | | > |
| 7 | **Opłata [zł]** | 1564 | 1444 | 2000 | 3000 |
| 8 | **Najkorzystniejsza oferta [zł]** | | **1444** | | |

1. Wstaw pasek przewijania w komórce **C6**.
   **Deweloper** → **Wstaw** → **Kontrolki formularza** → **Pasek przewijania** → przeciągnij kursor w poziomie
2. W razie potrzeby zmień orientację paska.
3. Sformatuj właściwości paska – określ wartości minimalną (**0**), maksymalną (**100**), zmianę przyrostową (**1**) oraz łącze do komórki (**B6**), w której będzie pojawiać się wartość.

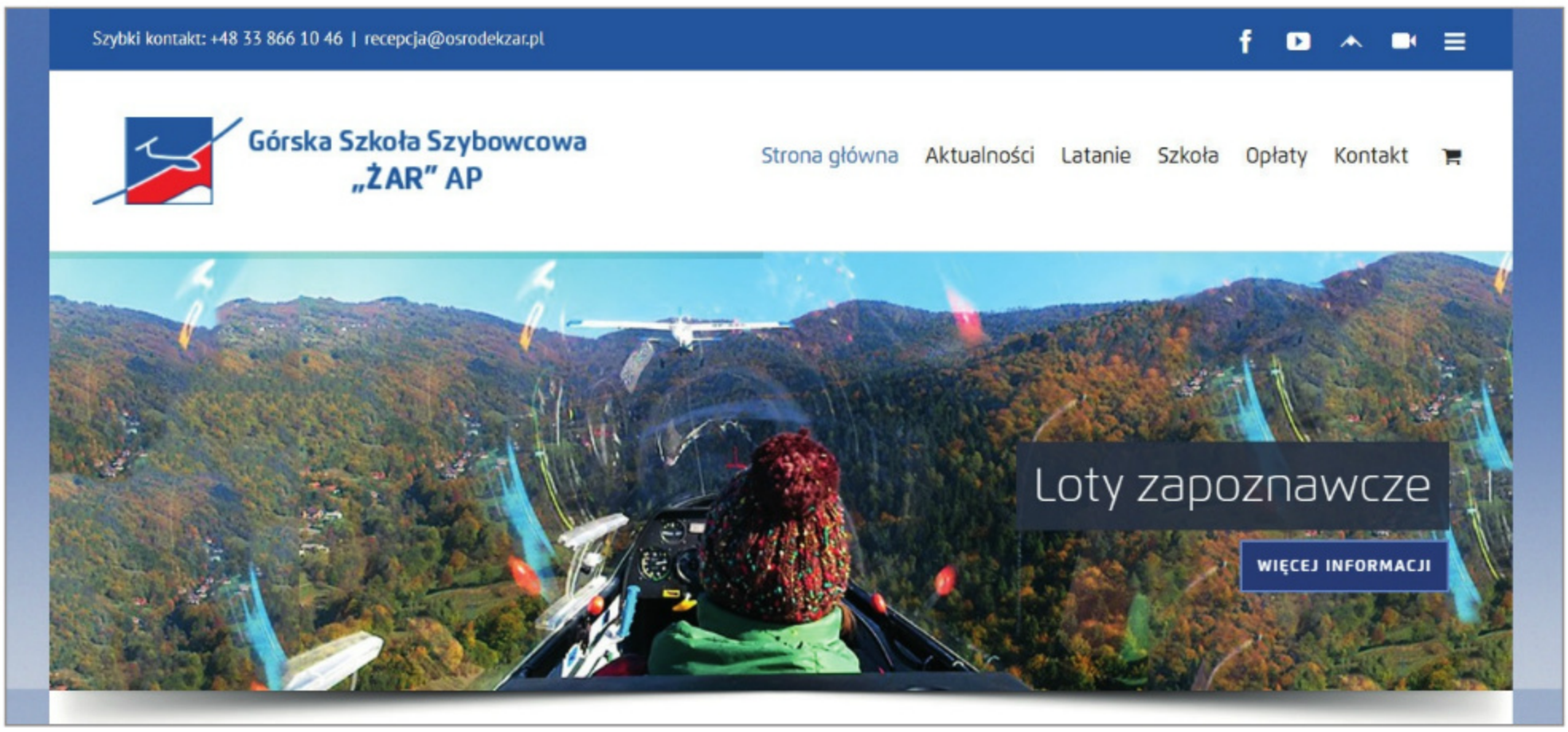


**W PRAKTYCE**

Arkusz można wykorzystać do symulacji kosztów internetu mobilnego niezbędnego w czasie wakacji (zanim porówna się oferty, należy sprawdzić, czy dany operator ma zasięg w miejscu pobytu!) albo do analizy wykorzystanych GB.

## Ćwiczenie 4. Kalkulator oferty z dodatkowymi opłatami

Poszukaj cen pakietów dla młodzieży w serwisie Górskiej Szkoły Szybowcowej „Żar" (**www.glidezar.com**) i sprawdź, ile będzie kosztował dwugodzinny lot szybowcem bez wykupionego pakietu. Uwzględnij opłatę za holowanie na 600 m (opłata za wysokość, jaką osiągnął zespół złożony z samolotu holującego i szybowca) oraz opłatę za nadzór (opłata za lot, niezależna od liczby godzin).




## PODSUMOWANIE

- Rzeczowa analiza ofert, traktowana jako problem algorytmiczny (formalny zapis problemu, wybór algorytmu rozwiązania, komputerowa realizacja, testowanie), umożliwia łatwy wybór najkorzystniejszej oferty.
- Podczas budowania modelu stosuje się pewne uproszczenia – pozwala to szybko otrzymać wynik, ale w niektórych przypadkach może negatywnie wpłynąć na ten wynik.
- Wprowadzenie nazw komórek i ich zakresów ułatwia analizę problemu – można wówczas posługiwać się nazwami zamiast adresami.
- Czytelnie zaprojektowany interfejs pozwala szybko zmieniać dane.

## PYTANIA SPRAWDZAJĄCE

1. Jakie kroki trzeba wykonać, aby w arkuszu kalkulacyjnym stworzyć symulator służący do porównywania ofert?
2. Dlaczego warto stosować nazwy komórek?
3. Dlaczego warto zastosować pasek przewijania do zmiany danych?

## PYTANIA QUIZOWE

1. W komórce **B1** wpisano kurs dolara amerykańskiego i nazwano ją **kurs**. Uczniowie szkoły szybowcowej zrobili zestawienie cen towarów, które chcieli zamówić w sklepie. Ceny podano w dolarach amerykańskich. Która formuła przeliczy prawidłowo cenę wpisaną w komórce **E2** w dolarach na złote?
   **a.** `=nazwa.kurs*E2`
   **b.** `=nazwa(B1)*E2`
   **c.** `=kurs*E2`
   **d.** `=E2/kurs`
2. Które dokończenie poniższego zdania jest prawdziwe?
   Aby odwołać się do poszczególnych komórek w arkuszu,
   **a.** zawsze wskazuje się adres.
   **b.** podaje się nazwę komórki lub zakresu.
   **c.** można podawać adres lub nazwę.
   **d.** można podać nazwę tylko wtedy, gdy nie używa się adresów.
3. Jaką orientację może mieć pasek przewijania?
   **a.** Tylko poziomą.
   **b.** Tylko pionową.
   **c.** Pod kątem 45 stopni.
   **d.** Poziomą lub pionową.

## ZADANIA DODATKOWE

### Zadanie 1. Telefonia komórkowa

Przeanalizuj oferty telefonii komórkowej i sporządź arkusz wyboru najlepszej oferty w zależności od liczby minut wykorzystanych na rozmowy oraz liczby gigabajtów w pakiecie internetowym.

### Zadanie 2. Wypożyczalnia rowerów

Opracuj symulator pozwalający wybrać wypożyczalnię rowerów w zależności od ceny za czas przejażdżki. Porównaj co najmniej dwie oferty. Wykorzystaj suwak do zmiany liczby minut.



# 5. Zakupy grupowe z arkuszem

- Przygotowanie arkusza do pracy grupowej
- Tworzenie list rozwijanych
- Formatowanie warunkowe

## ZADANIE NA START

**Jak kupić taniej**

Grę planszową *Chińczyk* możesz kupić w sklepie internetowym za 29,90 zł. Koszt dostawy wyniesie 8,90 zł. Ile zaoszczędzisz, jeśli kupisz tę samą grę za pośrednictwem platformy zakupów zbiorowych z rabatem 28 % i bezpłatną dostawą?



## ARKUSZ KALKULACYJNY JAKO PLATFORMA ZAKUPOWA

Kooperatywy to stowarzyszenia osób robiących wspólne zakupy m.in. za pomocą specjalnie do tego celu utworzonych platform (np. **https://kooperatywni.pl**), które dają możliwość składania zamówienia za pośrednictwem udostępnionego linku. Dzięki temu kupujący mogą uzyskać – w zależności od wielkości zamówienia – rabaty i inne bonusy, np. w postaci bezpłatnej dostawy.

Odpowiednio przygotowany arkusz kalkulacyjny również może stać się narzędziem do organizowania wspólnych zakupów. Operacje wykonywane w arkuszu przez kilka osób jednocześnie zazwyczaj działają prawidłowo, chociaż zdarzają się „nadpisania" danych. Historia zmian daje możliwość prześledzenia zmian w dokumencie.

Aby zorganizować np. zbiorowe zakupy gier przy użyciu arkusza kalkulacyjnego, trzeba:

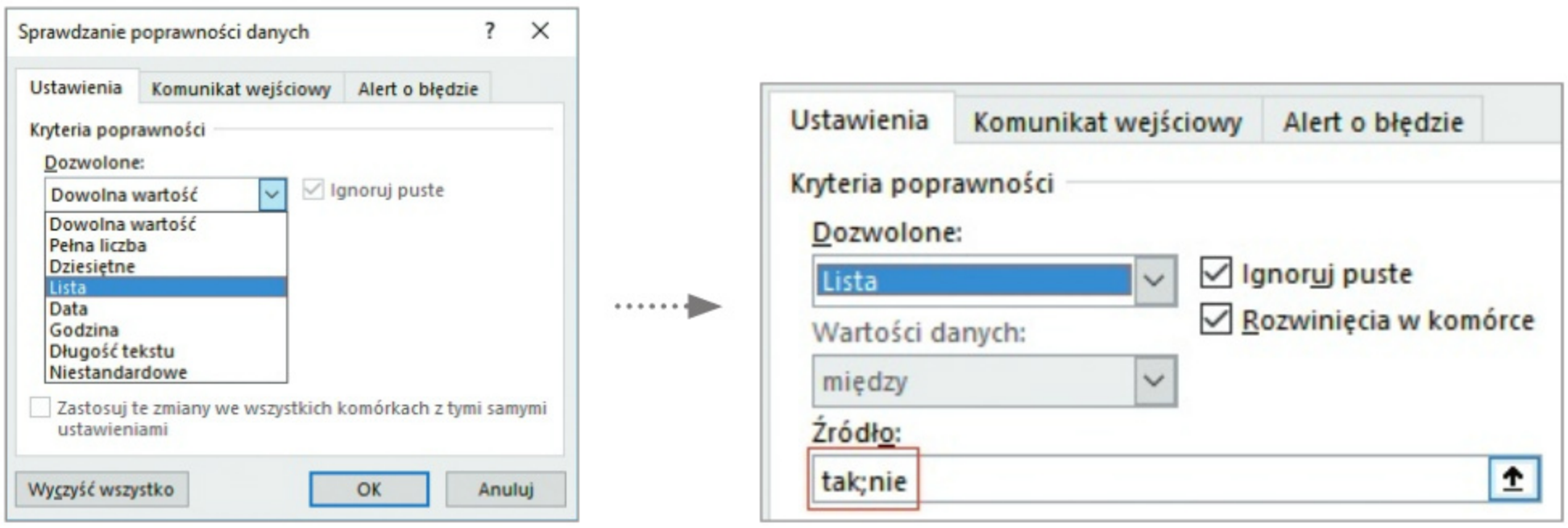
- przygotować w arkuszu spis gier oferowanych przez daną firmę (dany sklep);
- dodać informację o kategorii wiekowej dla każdej gry za pomocą listy rozwijanej i oznaczyć ją odpowiednim kolorem za pomocą formatowania warunkowego;
- dodać informację o cenie każdej gry;
- wprowadzić formuły, za pomocą których arkusz będzie obliczał rabat, podsumowywał zakupy każdej z osób osobno i wszystkich razem – potrzebne tu będą funkcje matematyczne oraz formuły warunkowe;
- umożliwić zamawianie gier znajomym – arkusz utworzony na dysku lokalnym można wyeksportować do Arkusza Google lub Office 365; link do udostępnionego arkusza można wysłać e-mailem albo udostępnić w portalu społecznościowym.

**W PRAKTYCE**

Zapamiętanie lub przekazanie komuś adresu strony internetowej w chmurze nie jest proste, ponieważ adres zazwyczaj jest bardzo długi. W internecie można znaleźć wiele serwisów oferujących bezpłatną **usługę skracania linków**. Za opłatą lub po rejestracji niektóre serwisy umożliwiają również wybór ciągu znaków, co pozwala zbudować skrót łatwy do zapamiętania.

## TWORZENIE LISTY ROZWIJANEJ

Dane wpisywane do arkusza nierzadko zawierają kilka wartości. W takim wypadku dobrze jest ograniczyć je wyłącznie do tych niezbędnych i utworzyć z nimi listę rozwijaną. W tym celu w programie Excel należy zaznaczyć odpowiednie komórki, a następnie wybrać **Dane → Narzędzia danych → Poprawność danych**. Wyświetli się okno **Sprawdzanie poprawności danych**. Na karcie **Ustawienia** w polu **Dozwolone** należy wybrać pozycję **Lista**, a w pole **Źródło** wpisać wartości oddzielone średnikami.




**Rys. 1.** Tworzenie listy rozwijanej z wartościami „tak” i „nie” w programie Excel

## Ćwiczenie 1. Lista rozwijana z kategorią wiekową

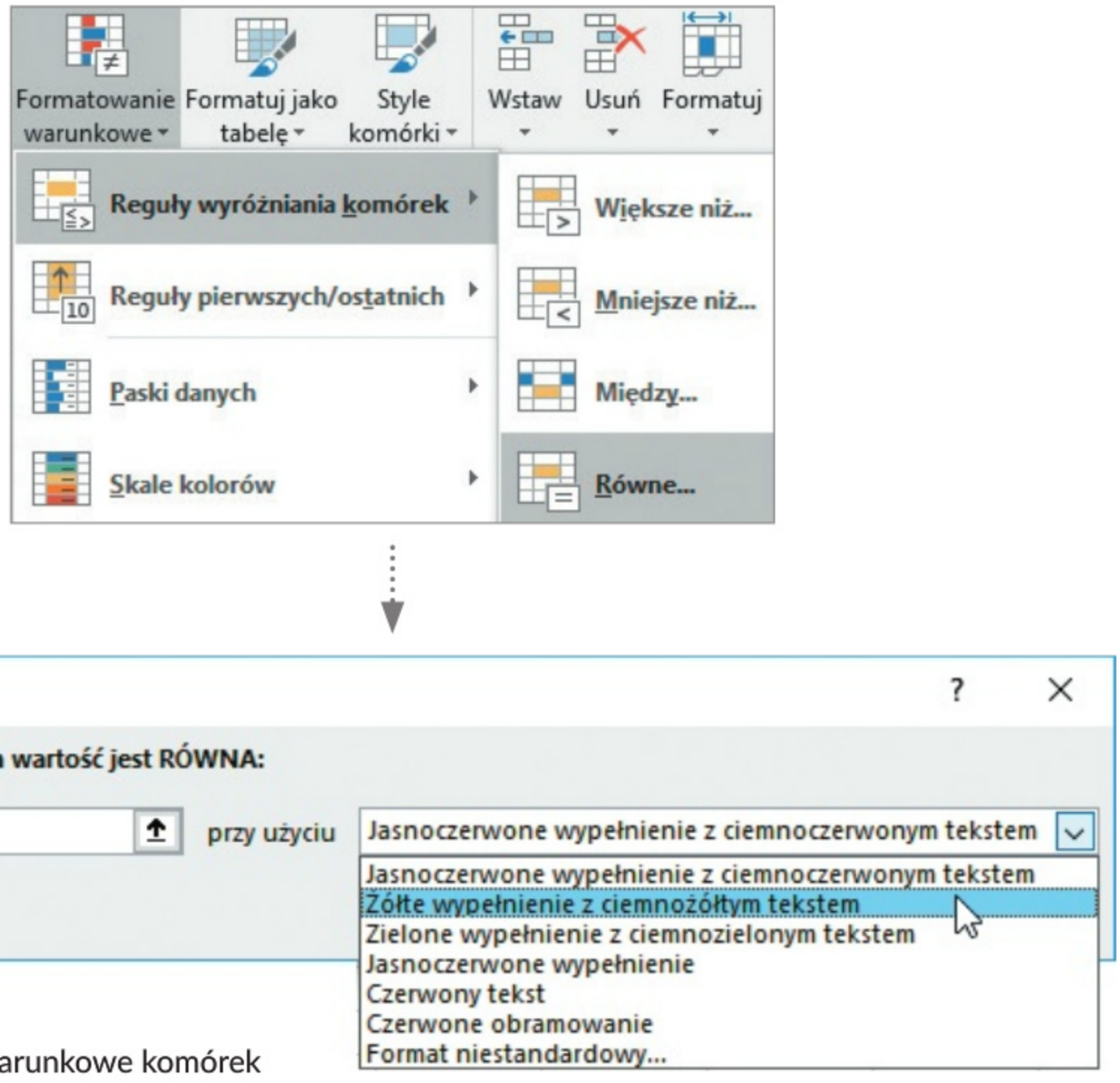
Przygotuj tabelę zawierającą spis gier planszowych, a następnie uzupełnij kolumnę **Wiek** za pomocą listy rozwijanej z następującymi kategoriami wiekowymi: 10+, 12+, 14+ (wzór poniżej).

| | A | B | C |
|---|---|---|---|
| 1 | **Lp.** | **Gra planszowa** | **Wiek** |
| 2 | 1 | 7 cudów świata | 10+ |
| 3 | 2 | Atak zombie | 12+ |
| 4 | 3 | Augustus | 10+ |
| 5 | 4 | Carcassonne. Hrabia, król i rze | 10+ 12+ 14+ |
| 6 | 5 | Coimbra | |
| 7 | 6 | Cthulhu Światy | |
| 8 | 7 | CV Pocket | |

1. Zaznacz komórki w kolumnie **Wiek**, a następnie otwórz okno **Sprawdzanie poprawności danych**.
   **Dane → Narzędzia danych → Poprawność danych**
2. W oknie **Sprawdzanie poprawności danych** w polu **Dozwolone** wybierz pozycję **Lista**.
   **Ustawienia → Dozwolone → Lista**
3. W pole **Źródło** wpisz oddzielone średnikami wartości: 10+, 12+, 14+.
4. Zatwierdź operację przyciskiem **OK**.
5. Rozwiń listę w każdej komórce i wybierz odpowiednią kategorię wiekową.

## FORMATOWANIE WARUNKOWE

Dzięki formatowaniu warunkowemu możliwe staje się stosowanie wybranego formatowania komórek na podstawie określonych warunków, takich jak powtarzające się wartości, wartości spełniające pewne kryteria itp. Aby zastosować tę funkcję arkusza, należy zaznaczyć odpowiednie komórki, a następnie wybrać **Narzędzia główne → Style → Formatowanie warunkowe → Reguły wyróżniania komórek**. Format zawartości wyróżnionych komórek wybiera się z dostępnej listy lub ustala samodzielnie za pomocą opcji **Format niestandardowy…**




**Rys. 2.** Formatowanie warunkowe komórek

## Ćwiczenie 2. Formatowanie warunkowe komórek

Uzupełnij tabelę z ćwiczenia 1. Zastosuj formatowanie warunkowe i ustal zróżnicowane kolory komórek umieszczonych obok gier przeznaczonych dla graczy w wieku powyżej 10, 12 i 14 lat (wzór poniżej).

| | A | B | C |
|---|---|---|---|
| 1 | **Lp.** | **Gra planszowa** | **Wiek** |
| 2 | 1 | 7 cudów świata | 10+ |
| 3 | 2 | Atak zombie | 12+ |
| 4 | 3 | Augustus | 10+ |
| 5 | 4 | Carcassonne. Hrabia, król i rzeka | 10+ |
| 6 | 5 | Coimbra | 14+ |
| 7 | 6 | Cthulhu Światy | 14+ |
| 8 | 7 | CV Pocket | 10+ |

1. Zaznacz komórki w kolumnie **Wiek**, a następnie ustal opcję **Reguły wyróżniania komórek**.

   Narzędzia główne → Style → Formatowanie warunkowe → Reguły wyróżniania komórek

2. Wybierz opcję **Równe...**, wpisz odpowiednią wartość, ustal sposób formatowania i zatwierdź operację przyciskiem **OK**.

3. Powtórz powyższe operacje dla pozostałych kategorii wiekowych.

**WIEM WIĘCEJ**

Podczas formatowania warunkowego warto korzystać z **Reguł pierwszych/ostatnich** do wyróżniania zestawu danych. Pozwalają one szybko znaleźć szukane wartości.

W zestawieniu pokazanym w tabeli obok wyróżniono np. wszystkie ceny powyżej średniej.

| | A | B |
|---|---|---|
| 1 | **Towar** | **Cena [zł]** |
| 2 | agrest | 3,90 |
| 3 | arbuzy | 1,50 |
| 4 | borówki | 26,00 |
| 5 | brzoskwinie | 14,30 |
| 6 | gruszki | 5,45 |
| 7 | jabłka | 2,50 |

### FUNKCJA SUMA.ILOCZYNÓW

Funkcja matematyczna **SUMA.ILOCZYNÓW** mnoży odpowiadające sobie elementy w podanych tablicach i zwraca sumę tych iloczynów. Musi mieć podane co najmniej dwie tablice tego samego rozmiaru. Wszystkie te pozycje tablicy, które nie są liczbami, traktuje tak, jakby były zerami.

**ZAPAMIĘTAJ!**

```
=SUMA.ILOCZYNÓW(zakres1;[zakres2];...)
```

## Ćwiczenie 3. Podsumowanie wydatków

Rozbuduj arkusz z ćwiczenia 2 tak, aby każdy, kto ma do niego dostęp, mógł zamawiać gry (wzór poniżej).

| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | **Lp.** | **Gra planszowa** | **Wiek** | **Cena brutto** | **Franek** | **Nina** | **Adam** |
| 2 | 1 | 7 cudów świata | 10+ | 114,00 zł | | | |
| 3 | 2 | Atak zombie | 12+ | 129,99 zł | 1 | | |
| 4 | 3 | Augustus | 10+ | 82,00 zł | | | 2 |
| 5 | 4 | Carcassonne. Hrabia, król i rzeka | 10+ | 45,00 zł | | | |
| 6 | 5 | Coimbra | 14+ | 199,95 zł | | 1 | |
| 7 | 6 | Cthulhu Światy | 14+ | 71,00 zł | | 1 | |
| 8 | 7 | CV Pocket | 10+ | 39,99 zł | | | |
| 9 | | | | | 129,99 zł | 270,95 zł | 164,00 zł |

1. Dopisz cenę brutto każdej gry.
2. Zrób miejsce na wpisywanie imion osób zamawiających oraz liczby zamawianych gier, a następnie uzupełnij trzy kolumny (od **E** do **G**).
3. Pod imieniem każdej osoby zrób podsumowanie wartości dokonanych zakupów – wykorzystaj funkcję **SUMA.ILOCZYNÓW** (dla kolumny **Franek** argumentami funkcji są ceny brutto i zamówienia Franka).

   `=SUMA.ILOCZYNÓW($D$2:$D$8;E2:E8)`
4. Podsumuj wartość pozostałych zamówień – skopiuj formułę do kolejnych komórek.

## Ćwiczenie 4. Zestawienie wydatków z rabatem

Uzupełnij arkusz z ćwiczenia 3 (wzór poniżej). Uwzględnij kolejne progi rabatów: standardowo 10 % ceny brutto; 20 % ceny brutto dla wartości zakupów brutto co najmniej 500 zł; 28 % ceny brutto dla wartości zakupów brutto co najmniej 1500 zł.

| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | **Lp.** | **Gra planszowa** | **Wiek** | **Cena brutto** | **Franek** | **Nina** | **Adam** |
| 2 | 1 | 7 cudów świata | 10+ | 114,00 zł | | | |
| 3 | 2 | Atak zombie | 12+ | 129,99 zł | 1 | | |
| 4 | 3 | Augustus | 10+ | 82,00 zł | | | 2 |
| 5 | 4 | Carcassonne. Hrabia, król i rzeka | 10+ | 45,00 zł | | | |
| 6 | 5 | Coimbra | 14+ | 199,95 zł | | 1 | |
| 7 | 6 | Cthulhu Światy | 14+ | 71,00 zł | | 1 | |
| 8 | 7 | CV Pocket | 10+ | 39,99 zł | | | |
| 9 | | | | | 129,99 zł | 270,95 zł | 164,00 zł |
| 10 | | **Razem** | 564,94 zł | | | | |
| 11 | | **Rabat** | 20% | | | | |
| 12 | | **Kwota po rabacie** | 451,95 zł | | | | |

1. Wpisz formułę obliczania rabatu.

   `=JEŻELI(C10>1500;28%;JEŻELI(C10>500;20%;10%))`

2. Wpisz formułę obliczania kwoty po rabacie.
   `=C10-C11*C10`

3. Zastosuj odpowiednie formaty komórek liczbowych – procenty i złote.

### Ćwiczenie 5. Podsumowanie zamówienia

Podsumuj liczbę zamówionych gier – dla każdej wpisanej osoby powinna się wyświetlić kwota do zapłaty z uwzględnieniem rabatu.

## PODSUMOWANIE

- Największą zaletą korzystania ze współdzielonego arkusza jest możliwość śledzenia zmian na bieżąco. Gdy jeden użytkownik wprowadza zmiany, inni natychmiast widzą efekt.
- Wykorzystanie rozwijanych list sprawia, że dane są poprawnie wprowadzane – użytkownik może podać tylko dane z listy.
- Formatowanie warunkowe przekłada się na zwiększoną czytelność arkusza, pozwala na szybkie zauważenie pewnych wartości.
- Dzięki korzystaniu z wbudowanych funkcji, takich jak **SUMA.ILOCZYNÓW**, przygotowanie arkusza okazuje się łatwiejsze.

## PYTANIA SPRAWDZAJĄCE

1. Jakie są etapy przygotowywania arkusza do wspólnych zakupów?
2. Jakie są plusy i minusy korzystania z arkusza współdzielonego?
3. Dlaczego stosuje się listy rozwijane z danymi, zamiast samemu wpisywać wartości?

## PYTANIA QUIZOWE

1. Do czego służy funkcja **SUMA.ILOCZYNÓW**?
   **a.** Mnoży odpowiadające sobie elementy w podanych tablicach i zwraca ich największą wartość.
   **b.** Odejmuje odpowiadające sobie składniki w podanych tablicach i zwraca ich iloczyn.
   **c.** Mnoży odpowiadające sobie elementy w podanych tablicach i zwraca sumę ich iloczynów.
   **d.** Dodaje odpowiadające sobie składniki w podanych tablicach i zwraca ich iloczyn.

2. Którą z poniższych czynności można wykonać za pomocą listy rozwijanej w komórkach arkusza?
   **a.** Nadać poprawne nazwy komórkom.
   **b.** Ograniczyć wpisy w komórce do podanych wartości.
   **c.** Wprowadzić formatowanie warunkowe.
   **d.** Zastosować adresowanie bezwzględne.
3. Co staje się możliwe dzięki formatowaniu warunkowemu?
   **a.** Skrócone obliczenia.
   **b.** Stosowanie zaawansowanych formuł.
   **c.** Udostępnianie arkusza do edycji.
   **d.** Stosowanie wybranego formatowania na podstawie określonych warunków.

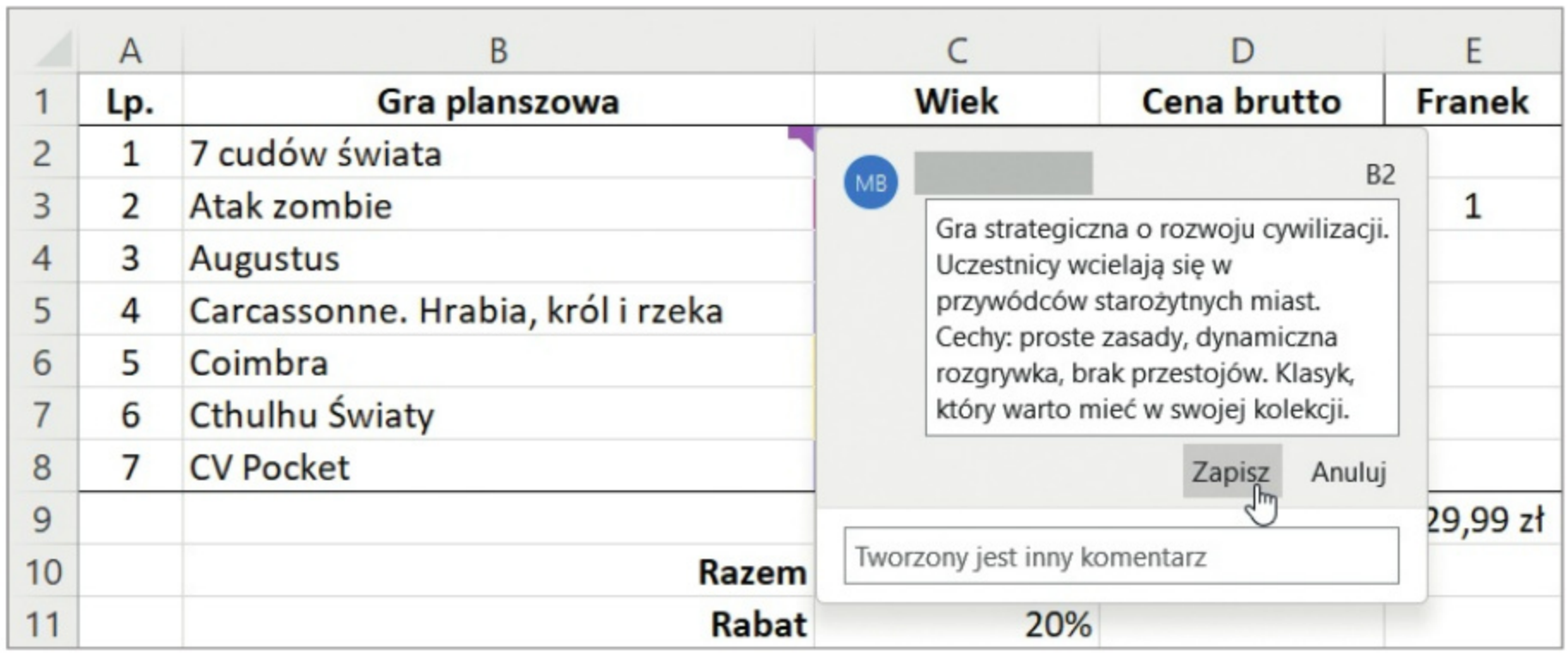
## ZADANIA DODATKOWE

### Zadanie 1. Cena z rabatem

Uzupełnij arkusz z ćwiczenia 5. Wstaw dodatkową kolumnę zawierającą ceny gier z uwzględnieniem aktualnego rabatu.

### Zadanie 2. Niewidoczne opisy

Uzupełnij arkusz z ćwiczenia 5 lub zadania 1. Przygotuj i wstaw skrócone opisy gier planszowych tak, aby nie było ich widać na stronie głównej.





# GRAFIKA RASTROWA

Jeden obraz wart jest więcej niż tysiąc słów. Zapotrzebowanie na obrazy tworzone cyfrowo od podstaw lub na bazie importowanych fotografii i rysunków widać właściwie w każdej dziedzinie. Na co dzień znajomość zagadnień dotyczących edycji grafiki może przydać się podczas:

- obróbki zdjęć publikowanych na Facebooku czy Instagramie,
- tworzenia kolaży zdjęć do fotokalendarzy,
- przygotowywania plakatu czy prezentacji.

# ZDJĘCIA – RETUSZ KOLORÓW

Fotografia to sztuka. Wymaga umiejętnej pracy światłem i sprzętem, wiedzy na temat kompozycji, kolorów i łączenia barw oraz znajomości programów do obróbki zdjęć. W GIMP-ie w menu **Kolor** znajdują się narzędzia, które pozwolą uzyskać pożądany efekt.

## BALANS BIELI

Balans bieli wpływa na to, czy zdjęcie wygląda naturalnie, czy też ma zniekształcone kolory. Przed wykonaniem fotografii warto ustawić w aparacie odpowiedni tryb oświetlenia, np. cień, światło dzienne, żarówek czy jarzeniowe, tak by obiekty, które są białe w rzeczywistości, były białe również na zdjęciu. Jeśli mimo wszystko efekt nie będzie zadowalający, należy dokonać retuszu.

GIMP: **Kolory → Automatycznie → Balans bieli**



Oryginał



Efekt

## NIEZBALANSOWANE KOLORY

Czasem idealne odwzorowanie kolorów może okazać się trudne lub wręcz niemożliwe, np. ze względu na zanieczyszczenie powietrza czy stopień nasłonecznienia. Jeśli nie pomoże ustawienie zaprogramowanych w aparacie trybów, zastosowanie filtrów polaryzacyjnych albo osłony na obiektywy, zafałszowane kolory należy poprawić w obróbce cyfrowej.

GIMP: **Kolory → Balans kolorów**



Oryginał



Efekt

## NIEDOŚWIETLENIE/ PRZEŚWIETLENIE

Aby zadbać o odpowiednią głębię ostrości, należy właściwie ustawić wielkość użytej przysłony, czas naświetlania i czułość ISO (niższa czułość = lepsza jakość). Jeśli zdjęcie jest prześwietlone do czystej bieli lub niedoświetlone do czystej czerni, nie można nic zrobić z utraconym szczegółem. Mniejsze niedoskonałości można skorygować w edytorze grafiki – w GIMP-ie należy wybrać **Kolory → Poziomy.**



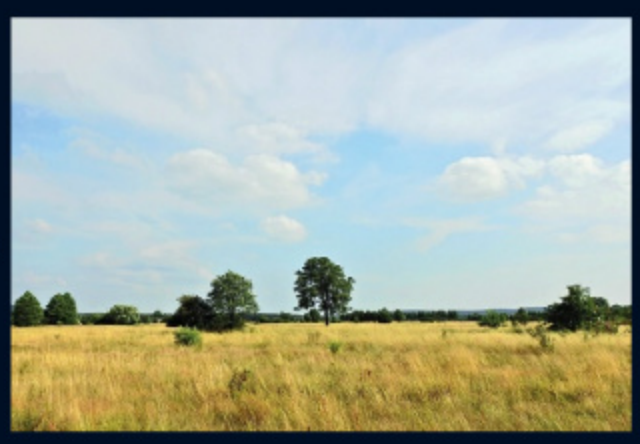
Zdjęcie niedoświetlone



Zdjęcie skorygowane



Zdjęcie prześwietlone



Zdjęcie skorygowane

### PRACA Z INFOGRAFIKĄ

1. Dlaczego należy korygować balans bieli?
2. Czy zdjęcia skorygowane z wykorzystaniem sztucznej inteligencji wyglądają naturalnie?

## MALOWANIE ŚWIATŁEM

To dosłowne tłumaczenie terminu „fotografia", który wywodzi się z języka greckiego i odwołuje się do techniki utrwalania obrazu na światłoczułym materiale. Wraz z rozwojem sztucznej inteligencji powstało wiele narzędzi do automatycznej korekcji fotografii – za pomocą różnych algorytmów można kolorować zdjęcia, usuwać plamy i rysy, poprawiać ostrość. Choć wiele z tych efektów można osiągnąć ręcznie w programie do obróbki grafiki, zastosowanie automatycznych narzędzi daje nowe możliwości.





Program Deoldify (**https://deoldify.ai/**) umożliwia automatyczne kolorowanie zdjęcia.



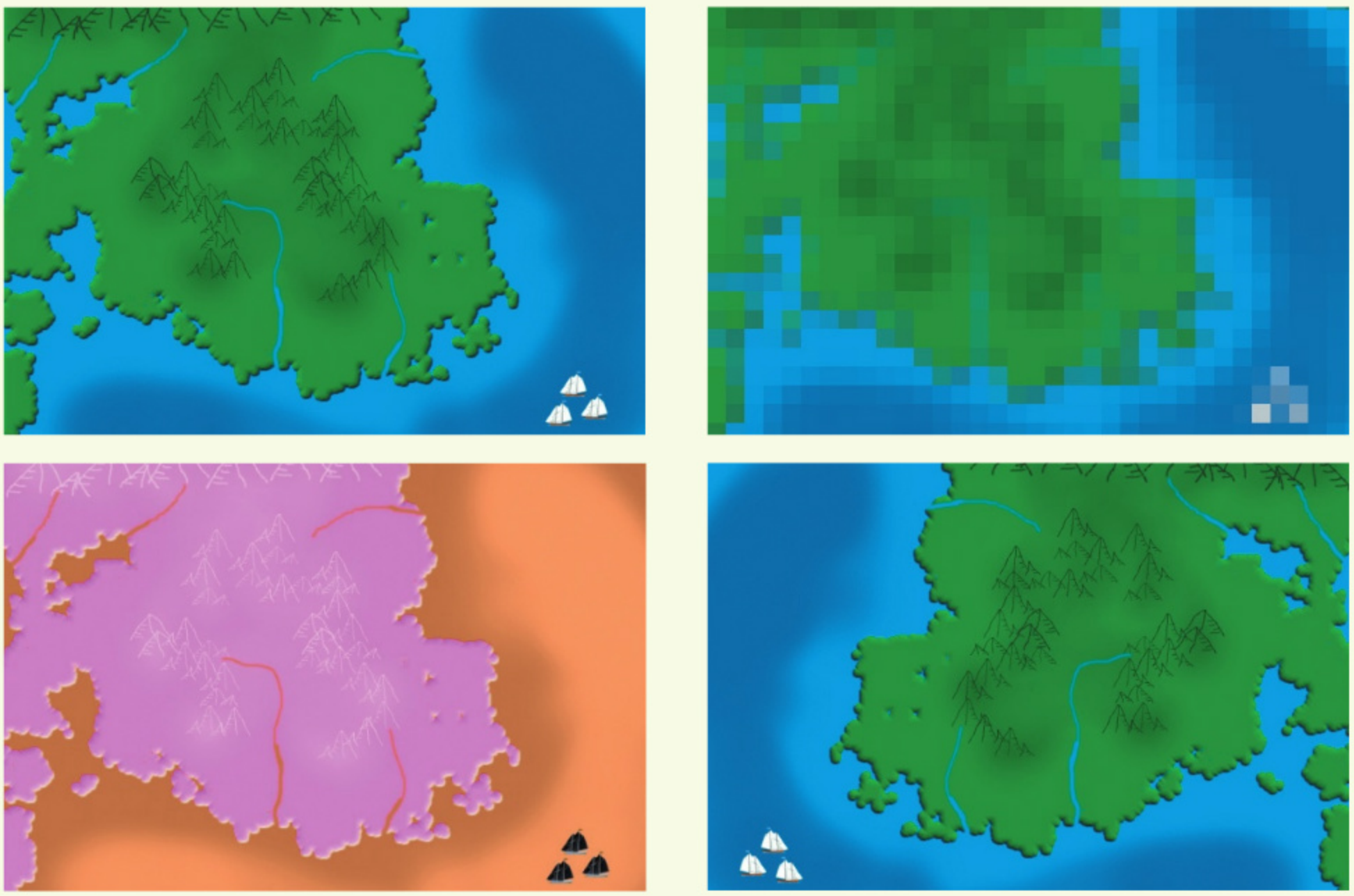
# 6. Podstawy edycji grafiki rastrowej

- Cechy charakterystyczne grafiki rastrowej
- Tworzenie i edytowanie rysunków w programie GIMP
- Dobór narzędzi w pracy nad projektem graficznym

## ZADANIE NA START

### Jaki efekt zastosowano

Edytor graficzny umożliwia wykonywanie różnych przekształceń. Przyjrzyj się rysunkowi oryginalnemu (lewy górny) i rysunkom zmodyfikowanym, a następnie zastanów się, w którym zastosowano kolejne efekty: odbicie pionowe, negatyw, pikselizację. Wyjaśnij, kiedy stosuje się takie efekty.



## GRAFIKA RASTROWA

Dane w komputerze są reprezentowane przez sekwencje bitów mających postać cyfr binarnych 0 i 1. System interpretuje ciąg zer i jedynek np. jako plik dźwiękowy, tekstowy, wideo czy graficzny, dzięki czemu użytkownik słyszy muzykę albo widzi tekst, film czy obraz.

W zależności od sposobu kodowania informacji o obrazie wyróżnia się grafikę wektorową (obiektową) i rastrową (bitmapową). W grafice wektorowej obraz jest opisywany za pomocą obiektów geometrycznych: figur (grafika 2D) lub brył (grafika 3D). W przypadku grafiki rastrowej obraz składa się z pojedynczych pikseli, które razem tworzą tablicę zwaną bitmapą.

Kolor każdego piksela można opisać za pomocą współrzędnych wybranego modelu przestrzeni barw. W przypadku modelu RGB barwa tworzona jest jako połączenie trzech wiązek światła – czerwonej (red), zielonej (green) i niebieskiej (blue). Każdy kolor może przyjmować wartości od 0 do 255.




**Rys. 1.** Tablica pikseli będących jednolitymi punktami o określonym kolorze

Ponieważ liczba pikseli tworzących obraz jest stała, właściwie niemożliwe jest skalowanie tego typu grafik bez utraty jakości. Przy powiększeniu edytor musi „uzupełnić" siatkę pikseli, dlatego krawędzie rozmywają się, maleje ostrość. Z kolei przy zmniejszaniu obrazu piksele zostają ściśnięte w mniejszej siatce.



**Rys. 2.** Widok bitmapy w skali 100 % i powiększonej do 1500 %

**W PRAKTYCE**

Grafika rastrowa doskonale sprawdza się w przypadku zdjęć oraz złożonych projektów fotorealistycznych, w których dużą wagę przykłada się do głębi i subtelności barw.

## GIMP – WPROWADZENIE

GIMP (GNU *Image Manipulation Program*) to bezpłatna aplikacja pozwalająca na edycję grafiki rastrowej w trybie offline w różnych systemach operacyjnych. Można ją pobrać ze strony **www.gimp.org/downloads**.

Wygląd edytora można dostosować do swoich preferencji. Aby to zrobić, należy:

- w menu **Edycja** wybrać opcję **Preferencje**,
- w menu **Interfejs** kliknąć **Motyw**,
- wskazać wybrany wygląd w oknie **Wybór motywu** po prawej stronie ekranu (może to być np. motyw **System**).

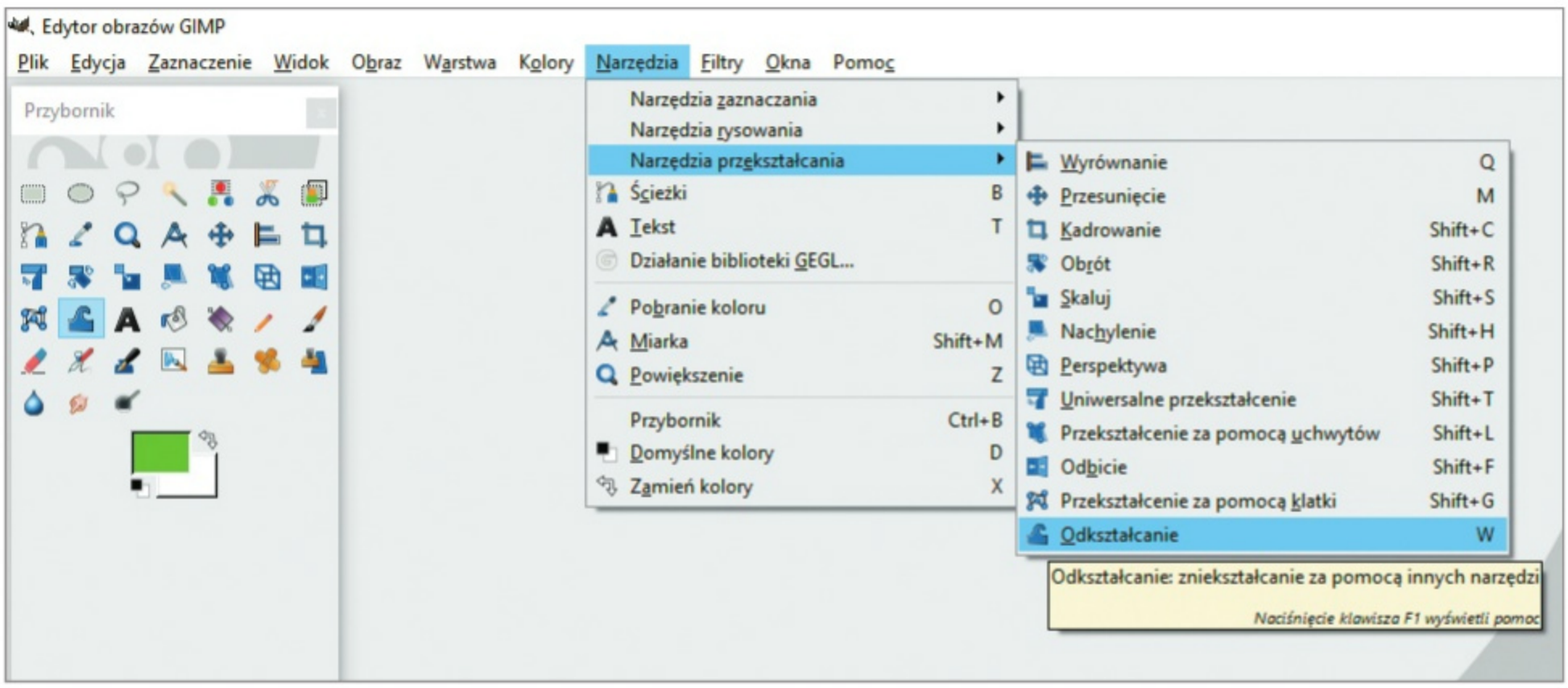
Nieco niżej w menu **Interfejs** znajduje się **Motyw ikon**, w którym można dopasować wygląd ikon w programie GIMP (np. motyw **Color**) w sposób analogiczny do opisanego powyżej.

Aby otworzyć istniejący plik zapisany na dysku, można kliknąć **Plik → Otwórz...**, wybrać odpowiedni plik albo otworzyć folder, w którym przechowywany jest dany plik, i przeciągnąć go do okna GIMP-a. Aby wczytać plik bezpośrednio z internetu, trzeba kliknąć **Plik → Otwórz położenie...** i wpisać adres URL.

W celu utworzenia nowego obrazu wystarczy kliknąć **Plik → Nowy...**, a następnie w oknie dialogowym podać parametry podstawowe i zaawansowane.

## NARZĘDZIA

Narzędzia w GIMP-ie umieszczone są w **Przyborniku** i w menu. Podstawowe narzędzia w większości programów do obróbki grafiki są zbliżone.




**Rys. 3.** Okno **Przybornika** i rozwinięte menu **Narzędzia** z zaznaczonym narzędziem **Odkształcanie**

Narzędzia zgromadzone w **Przyborniku** można pogrupować następująco:

- zaznaczanie – prostokątne, eliptyczne, swobodne (odręczne – lasso, różdżka, według koloru) i abstrakcyjne (inteligentne nożyce, pierwszy plan);
- rysowanie – wypełnienie kubełkiem, gradient, ołówek, pędzel, gumka, aerograf, stalówka, klonowanie, łatka, klon perspektywy, rozmywanie/wyostrzanie, rozsmarowywanie, rozjaśnianie/przyciemnianie;
- przekształcanie – przesunięcie, wyrównanie, kadrowanie, uniwersalne przekształcenie, obrót, skalowanie, nachylenie, przekształcenie za pomocą uchwytów, perspektywa, odbicie, przekształcenie za pomocą klatki, odkształcenie;
- pozostałe – ścieżki, pobranie koloru, powiększenie, miarka, tekst.

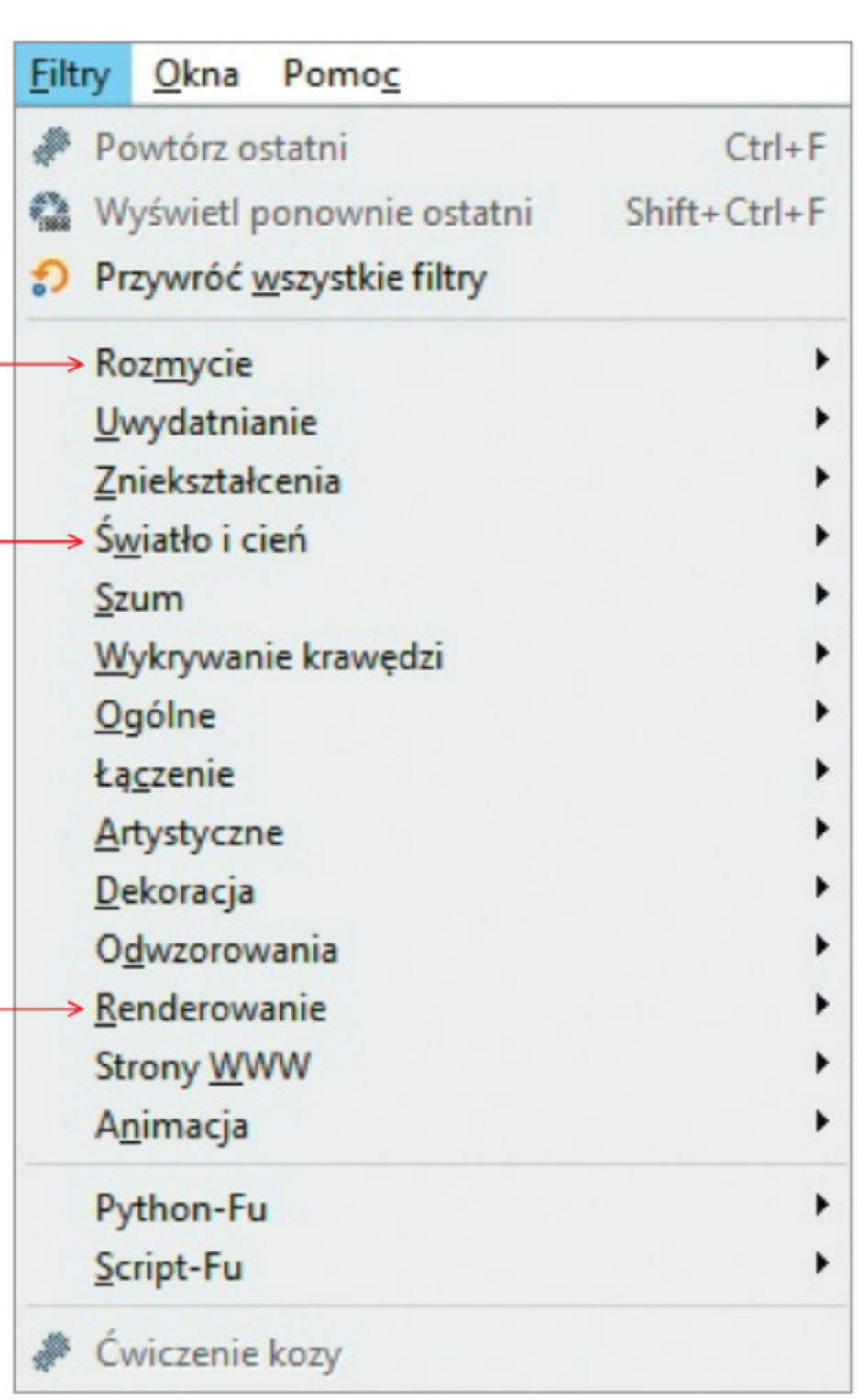
Menu programu znajduje się w górnej części okna edytora i pod prawym przyciskiem myszy. Dodatkowe opcje można znaleźć w tzw. dokowanych oknach dialogowych. Są tam pędzle, warstwy obrazu i historia zmian.

Narzędzia umieszczone w menu **Filtry** służą do modyfikacji obrazu według różnych parametrów. GIMP oferuje ich wyjątkowo dużo. Poza tym dostępne są wtyczki rozszerzające możliwości aplikacji.

**Rozmycie** pozwala m.in. na pikselizację, rozmycie ruchu (zwiększenie dynamiki obrazu) i rozmycie Gaussa (równomierne lub selektywne rozmycie obrazu).

**Światło i cień** pozwala m.in. dodać efekt błysku soczewki, rzucić cień albo uzyskać subtelny półprzezroczysty cień z obrysem (**Efekt Xach...**).

**Renderowanie** pozwala na utworzenie nowych obrazów – fraktali, szumów i deseni.




**Uwydatnianie** pozwala usuwać niedoskonałości – redukować szum, wygładzać i „odplamiać".

**Szum** pozwala m.in. losowo zniekształcić kolory (**Szum RGB...**) albo wybrać barwę i nasycenie (**Szum HSV...**), przesunąć lub wymienić piksele.

**Dekoracja** pozwala m.in. dodać ciekawe efekty typu mgliste chmury (**Fog...**), plama po kawie czy poszarpana krawędź albo przestrzenna ramka (**Dodaj fazę...**).

**Rys. 4.** Rozwinięta lista grup filtrów dostępnych w edytorze GIMP

**WIEM WIĘCEJ**

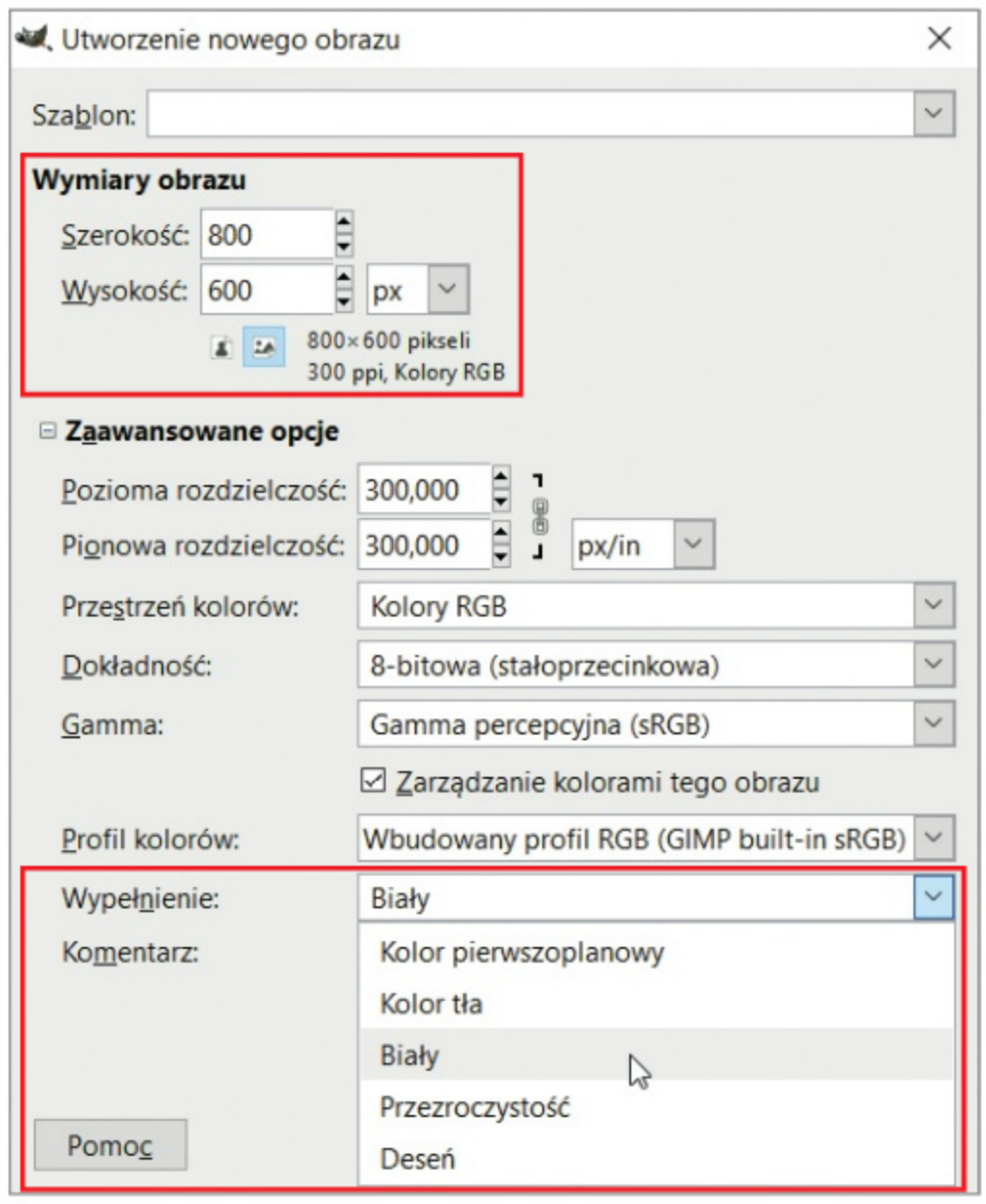
**Wtyczka** (ang. *plug-in*, *add-on*) to zewnętrzny program uruchamiany pod kontrolą głównej aplikacji. W GIMP-ie zainstalowano dużo tego typu programów – właściwie wszystko w menu **Filtry** jest wtyczką. Aby się o tym przekonać, wystarczy wybrać **Pomoc → Przeglądarka wtyczek**.

## Ćwiczenie 1. Przygotowanie kanwy

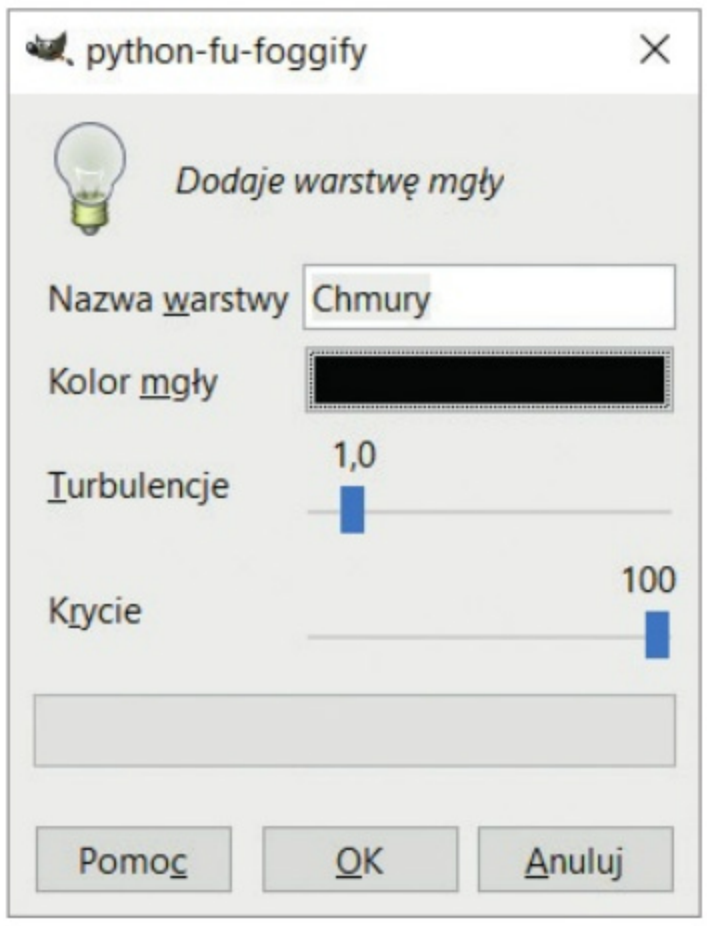
Przygotuj w edytorze GIMP dwukolorowe tło, które będzie stanowiło podstawę mapy (wzór poniżej).



1. Utwórz nowy plik o rozmiarze 800 × 600 px i białym tle.

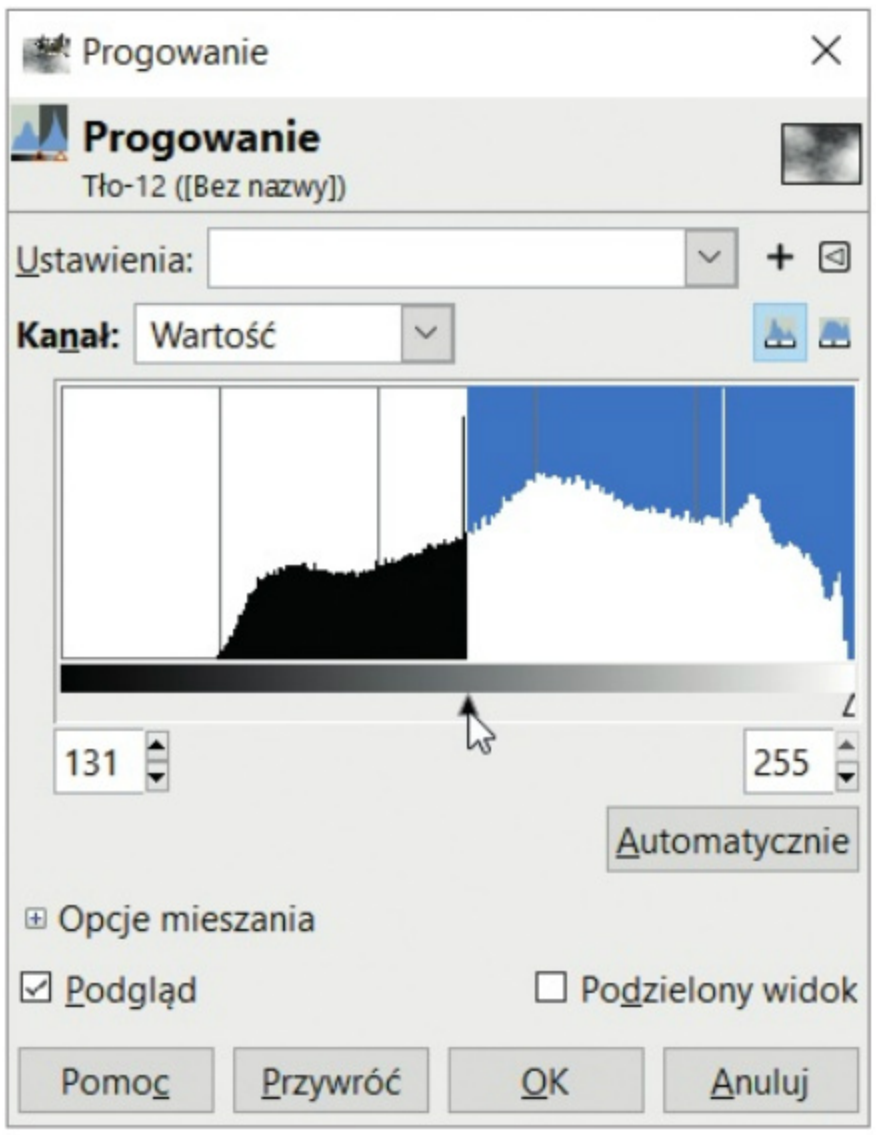


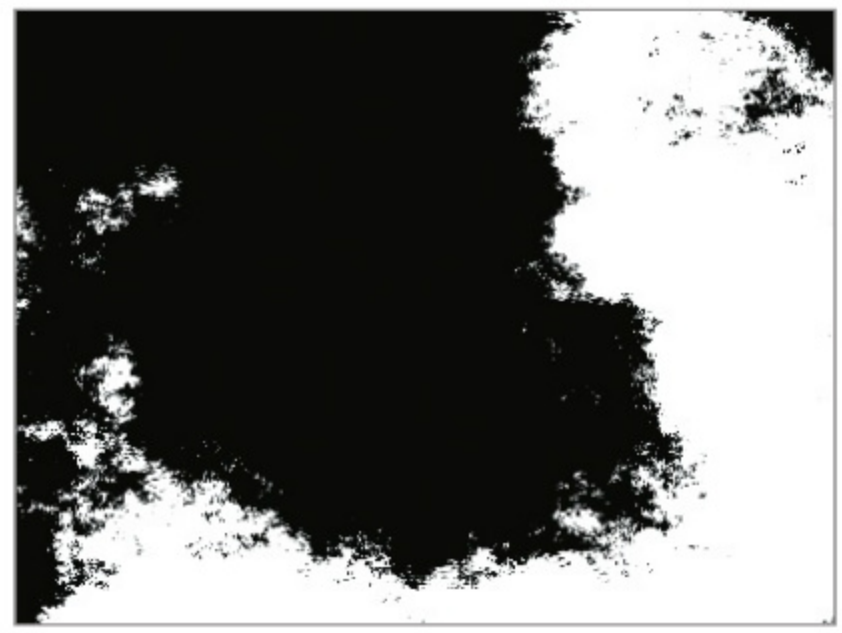
2. Wygeneruj losową mapę – wykorzystaj filtr **Mgła...** z grupy **Dekoracja**. Ustal kolor mgły na czarny, wartość turbulencji – na 1,0 (wyższa wartość oznacza większą wyrazistość mgły, mniejsza spowoduje efekt rozmycia), a krycie (stopień przezroczystości) – na 100 %.




3. Ponieważ domyślnie mgła jest tworzona na nowej warstwie, połącz warstwy w jedną – naciśnij prawy przycisk myszy i wybierz z menu podręcznego **Warstwa → Połącz w dół**.

4. Przekształć rysunek w czarno-biały obraz – wykorzystaj narzędzia **Progowanie** z menu **Kolory**. Dopasuj wartość progu tak, aby na płótnie były zarówno obszary białe, jak i czarne.



5. Wygładź postrzępiony brzeg – wybierz w **Przyborniku** narzędzie **Zaznaczenie według koloru**, zaznacz kolor czarny, a następnie w menu **Zaznaczenie** kliknij opcję **Powiększ...** i określ stopień powiększenia zaznaczenia.

6. Zamień kolor czarny na zielony – ustal aktywny kolor pierwszoplanowy, zaznacz w **Przyborniku** narzędzie **Wypełnienie kubełkiem** i kliknij zaznaczony obszar. Całe zaznaczenie wraz z powiększeniem zostanie wypełnione wybranym kolorem.

7. Zamień kolor biały na niebieski – ponownie ustal aktywny kolor pierwszoplanowy, zaznacz w **Przyborniku** narzędzie **Wypełnienie kubełkiem** i kliknij biały obszar.

8. Usuń zaznaczenie – w menu **Zaznaczenie** kliknij **Brak**.

9. Zapisz plik w domyślnym dla GIMP-a formacie XCF – w menu **Plik** wybierz opcję **Zapisz jako...**

## Ćwiczenie 2. Nadawanie tekstury

Przekształć dwukolorowe tło z ćwiczenia 1 w rysunek oddający charakterystykę terenu – nadaj linii brzegowej wrażenie trójwymiarowości, a następnie zróżnicuj kolorystycznie obszary zielone i niebieskie (wzór poniżej).



1. Uzyskaj efekt wypukłego brzegu – wybierz narzędzie **Zaznaczanie według koloru**, zaznacz obszar zielony, a następnie zaznacz filtr **Dodaj fazę...** z grupy **Dekoracja**. Zmień odpowiednio parametr **Grubość** lub pozostaw wartość domyślną.

2. Zróżnicuj kolorystycznie obszary zielone – pobierz pipetą kolor zielony i lekko go przyciemnij, a następnie wybierz z grupy **Narzędzia rysowania** narzędzie, które pozwoli na dodanie rozmytych plam, np. miękki pędzel (z kryciem ustawionym na maksymalnie 50 %) lub rozsmarowywanie.

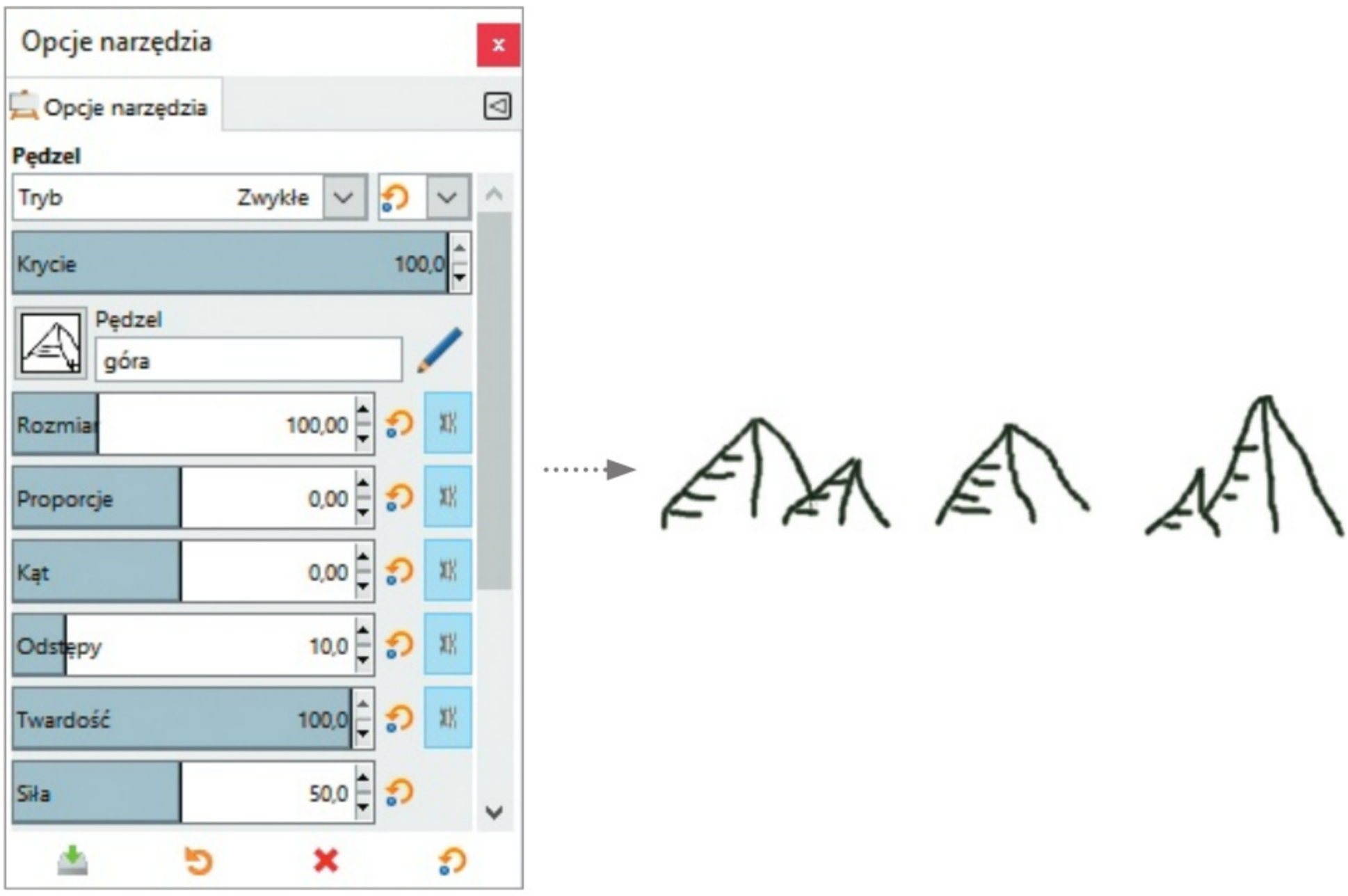
3. Zróżnicuj kolorystycznie obszary niebieskie.

4. Zapisz zmodyfikowany plik.

## TWORZENIE WŁASNYCH PĘDZLI

Standardowo pędzel ma kształt koła, można jednak wybrać inny kształt, np. kwadratu, gwiazdki lub… papryki. Ponadto warto nauczyć się tworzenia własnych pędzli, które uproszczą powielanie obiektu, ponieważ będą działać niczym stempel.

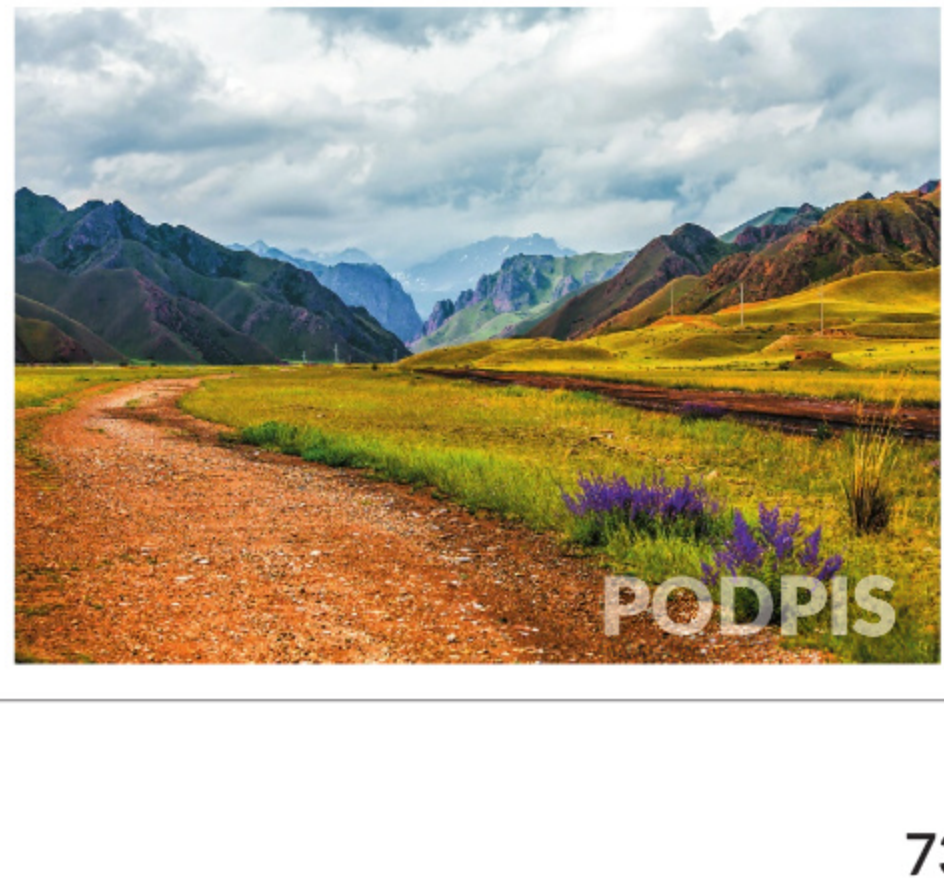
Nowy pędzel można przygotować jako mały rysunek na przezroczystym tle. Taki rysunek eksportuje się do formatu pędzla (**Plik → Wyeksportuj jako → Wybór typu pliku** (**Według rozszerzenia**) → **Pędzel programu GIMP**) i zapisuje w podkatalogu przeznaczonym dla pędzli (*brushes*) w katalogu własnym GIMP-a, np. C:\Program Files\GIMP 2\share\gimp\2.0\brushes (ta operacja może wymagać uprawnień administratora).




**Rys. 5.** Pędzle-stemple w kształcie gór według własnego projektu

### W PRAKTYCE

Aby ochronić fotografię lub grafikę opublikowaną w sieci przed wykorzystaniem bez zgody autora, można **zabezpieczyć obraz znakiem wodnym** w postaci podpisu. Należy przemyśleć formę i umiejscowienie sygnatury. Znak powinien znajdować się w miejscu unikatowym, tak by nie dało się go łatwo usunąć (tzn. nie na jednolitym tle).



## Ćwiczenie 3. Dodawanie szczegółów

Dodaj szczegóły do mapy przygotowanej w ćwiczeniu 2 – dorysuj rzeki, góry i żaglówki (wzór poniżej).



1. Utwórz własne pędzle-stemple.
2. Jeśli na rysunku ma się znaleźć wiele podobnych obiektów, przygotuj starannie jeden obiekt, a następnie go skopiuj (potem możesz nieznacznie zmodyfikować kopie).
3. Zapisz wprowadzone zmiany.

### PODSUMOWANIE

- W grafice rastrowej obraz składa się z pojedynczych kolorowych pikseli. Podczas różnych przekształceń, np. zmiany wymiarów, jakość takiego obrazu może się znacząco pogorszyć.
- Przed rozpoczęciem pracy nad projektem graficznym warto poznać podstawowe narzędzia edytora – odpowiedni dobór narzędzi ułatwi realizację celu i zapewni pożądany efekt.
- Filtr graficzny dodaje niestandardowe efekty w trakcie obróbki obrazu, np. filtr **Mgła...** z grupy **Dekoracja** pozwala uzyskać mglistą chmurę.
- GIMP pozwala tworzyć własne pędzle, które upraszczają powielanie obiektu, ponieważ działają niczym stemple.

### PYTANIA SPRAWDZAJĄCE

1. Czym jest grafika rastrowa?
2. Co odróżnia grafikę rastrową od wektorowej?
3. Jakie są zastosowania grafiki rastrowej?

## PYTANIA QUIZOWE

1. Jaki jest charakterystyczny format plików dla programu GIMP?
   **a.** XCF
   **b.** PNG
   **c.** WAV
   **d.** WEB
2. Jaki jest efekt progowania?
   **a.** Część rysunku staje się czerwona, druga część – zielona.
   **b.** Rysunek zamienia się w obraz czarno-biały bez kolorów pośrednich.
   **c.** Rysunek zamienia się w obraz czarno-biały z odcieniami szarości.
   **d.** Najjaśniejsze części rysunku stają się czarne.
3. Co powoduje zastosowanie filtru w edytorze grafiki?
   **a.** Poprawę jakości obrazu.
   **b.** Utratę ostrości.
   **c.** Modyfikację według określonego algorytmu.
   **d.** Usunięcie ciemnych miejsc i zastąpienie ich jaśniejszymi.

## ZADANIA DODATKOWE

### Zadanie 1. Czego brakuje na mapie?

Przyjrzyj się uważnie mapie z ćwiczenia 3 i dodaj nowe elementy. Postaraj się wykorzystać do tego różne narzędzia. Inspiracji możesz szukać m.in. na YouTubie – wiele osób umieszcza tam swoje graficzne poradniki, także dotyczące map. Na koniec poproś o ocenę pracy koleżanki i kolegów.

### Zadanie 2. Mapa atrakcji regionu

Na podstawie konturów uproszczonej mapy swojego regionu przygotuj mapę prezentującą atrakcje turystyczne okolicy. Zastanów się, jakie elementy warto na niej umieścić, aby podkreślić zalety regionu, a następnie dobierz odpowiednie narzędzia.

### Zadanie 3. Gra komputerowa

Odpowiednio zaprojektowana mapa może stanowić planszę w grze komputerowej. Przygotuj propozycję takiej planszy. Zwróć uwagę na parametry techniczne – wielkość planszy, kolorystykę całego projektu, proporcje między elementami ruchomymi i nieruchomymi.



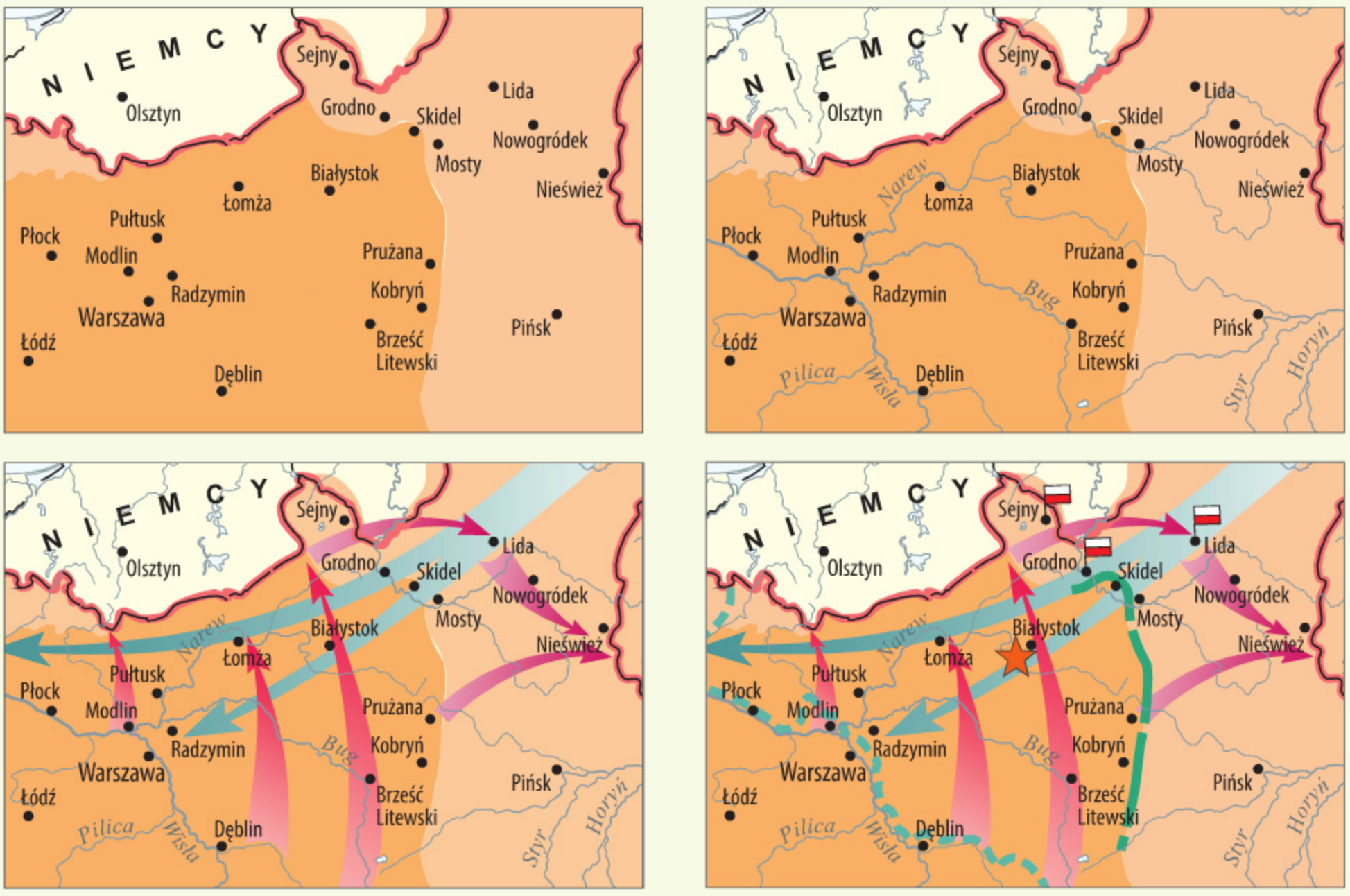
# 7. Praca na warstwach

- Zasady pracy na warstwach
- Wykorzystanie warstw do tworzenia i modyfikacji grafiki
- Formaty plików graficznych i ich zastosowanie

## ZADANIE NA START

### Różne wersje mapy

Na poniższych rysunkach przedstawiono ten sam fragment mapy Polski z czasów wojny polsko-bolszewickiej w 1920 roku, ale na każdym z nich umieszczono inne informacje. Jakie nowe elementy można znaleźć na poszczególnych rysunkach?



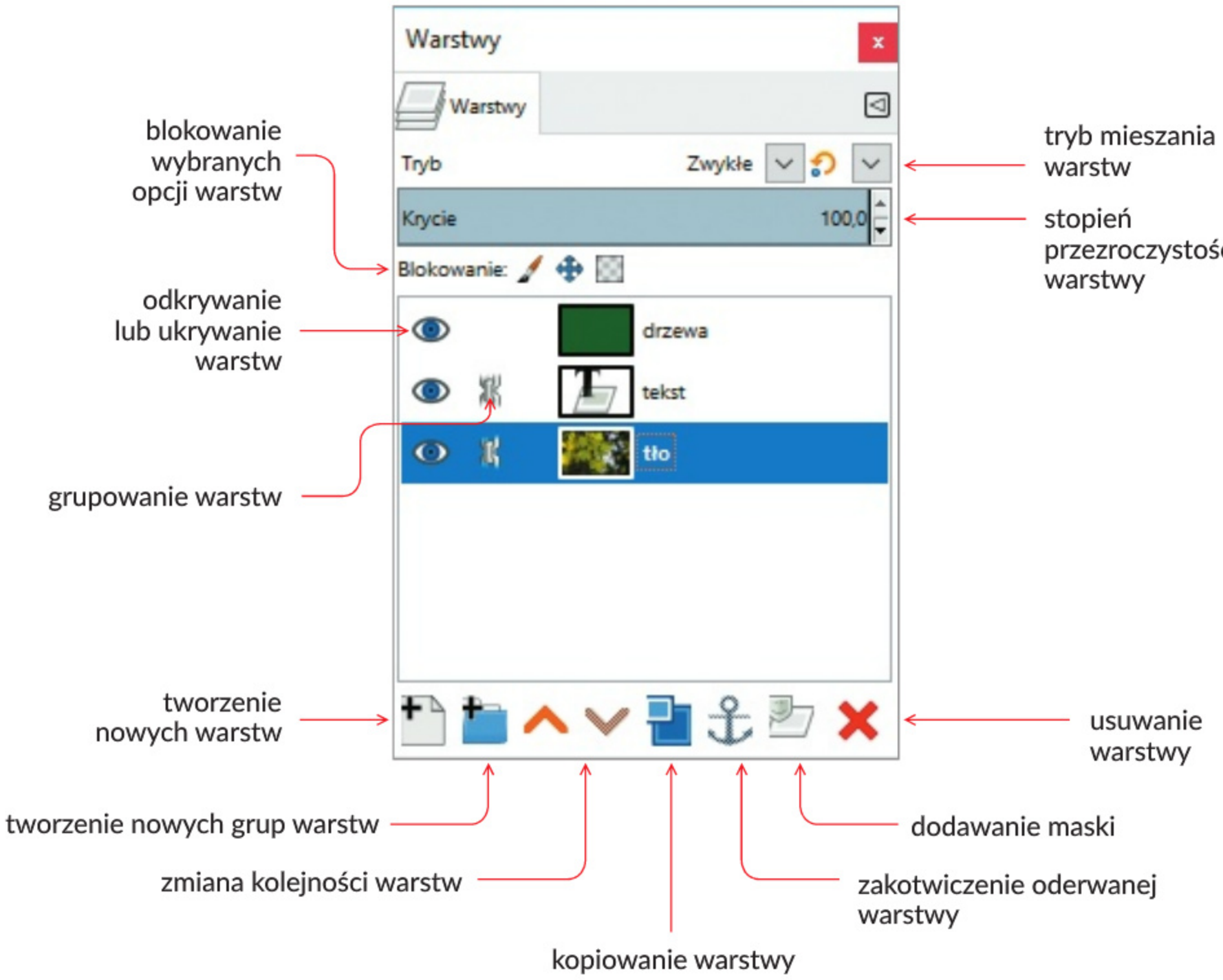

## PRACA NA WARSTWACH W PROGRAMIE GIMP

Warstwy można porównać do przezroczystych folii, które zawierają fragmenty obrazu i po odpowiednim nałożeniu na siebie tworzą całość. Każdy obraz może się składać z jednej lub wielu warstw. Ponieważ są to obiekty niezależne, można je dowolnie edytować (np. rysować na nich, wstawiać tło) i przesuwać względem siebie. Nowy plik ma tylko jedną warstwę, zwaną warstwą tła.

**WIEM WIĘCEJ**

Czasami edytor automatycznie tworzy nową warstwę. GIMP robi tak np. przy dodawaniu napisu czy wprowadzaniu efektu **Mgła...** z grupy filtrów **Dekoracja**.

Po instalacji programu GIMP okno dialogowe zawierające warstwy obrazu jest domyślnie otwarte. Jeśli jest zamknięte, można je otworzyć przez wybranie w menu sekwencji: **Okna → Dokowalne okna dialogowe → Warstwy**.




**Rys. 1.** Okno warstw w edytorze GIMP

**ZAPAMIĘTAJ!**

Główne **zasady pracy na warstwach**:

- warstwy położone wyżej w oknie warstw przesłaniają te, które są niżej;
- aktualnie zaznaczona warstwa jest edytowana;
- każda warstwa ma unikatową nazwę;
- rozmiary warstw nie muszą być takie same;
- warstwy spięte spinaczem można edytować jednocześnie.

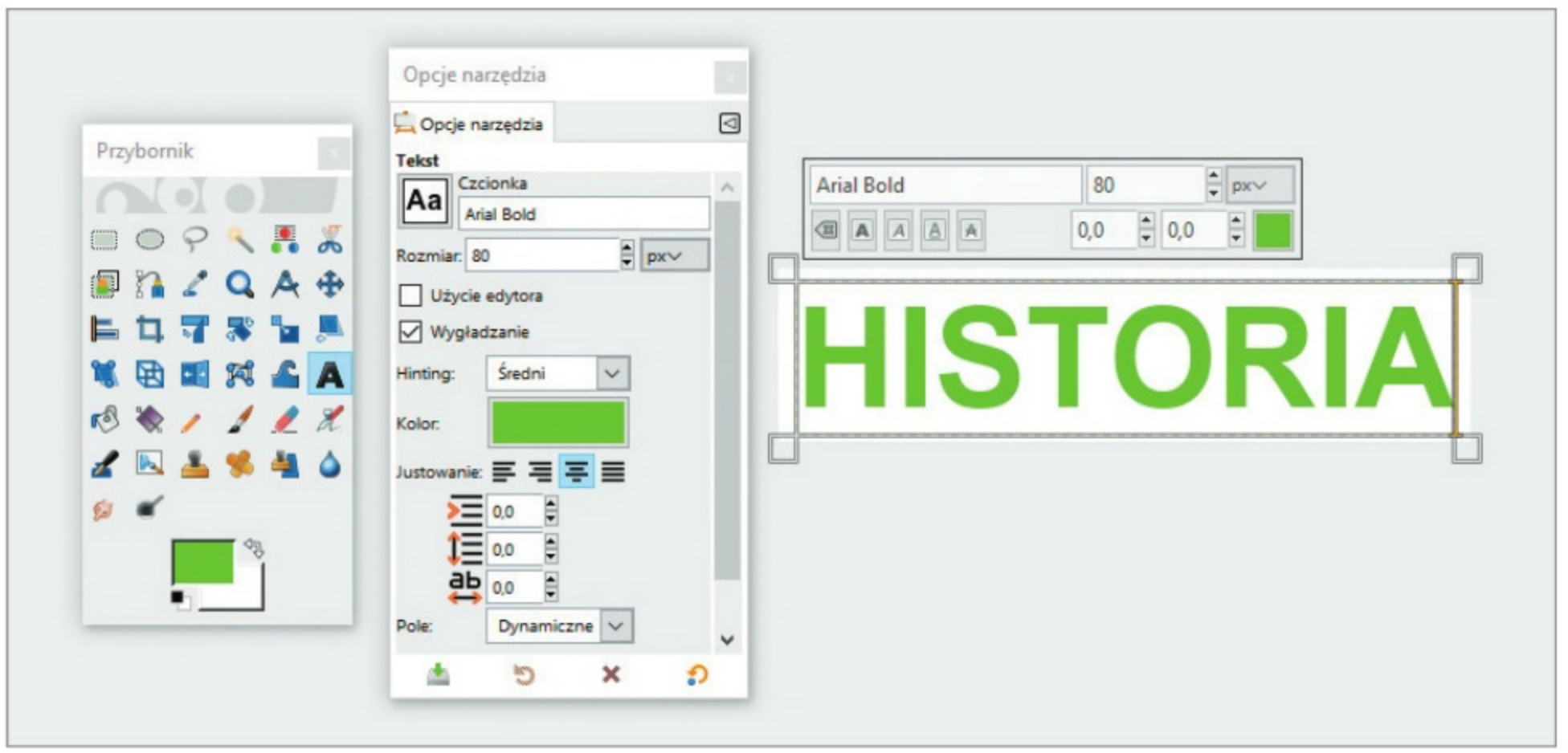
## Ćwiczenie 1. Tworzenie prostego napisu z cieniem

Utwórz w edytorze GIMP napis z efektem przesunięcia.




1. Utwórz nowy plik o rozmiarze 400 × 100 px i białym tle.

2. Wybierz w **Przyborniku** narzędzie **Tekst**, ustal parametry w opcjach narzędzia i wpisz tekst (powstanie nowa warstwa tekstowa o nazwie takiej jak wpisany tekst).




3. Zduplikuj warstwę tekstową.




4. Zmień kolor tekstu w starej warstwie na czarny – ukryj nową warstwę (odznacz „oko” w oknie warstw przy nowej warstwie), zaznacz w oknie warstw starą warstwę, zaznacz tekst na płótnie (np. za pomocą skrótu klawiaturowego **Ctrl+A**) i zmień kolor w opcjach narzędzia.

5. Stwórz efekt wypukłości – w oknie warstw zaznacz „oko” przy nowej warstwie, a potem – samą warstwę, następnie kliknij w **Przyborniku** narzędzie **Przesunięcie** i przeciągnij nową warstwę w prawo i lekko w dół.

6. Zapisz plik w formacie XCF, aby mieć dostęp do wszystkich warstw.

## Ćwiczenie 2. Tworzenie napisu na tle obrazka

Utwórz obraz zawierający napis **1920** na tle polskiej flagi.




1. Pobierz z serwisu Wikimedia Commons flagę, która będzie stanowiła tło.
2. Ustal parametry tekstu i napisz tekst.
3. Dodaj subtelny, półprzezroczysty efekt 3D – wykorzystaj **Efekt Xach...** z grupy filtrów **Światło i cień**.
4. Wyczyść zaznaczone piksele – w menu **Edycja** kliknij polecenie **Wyczyść**.
5. Zmień aktywny kolor narzędzia tekstowego, np. na granatowy.
6. Dodaj obrys napisu – w menu **Edycja** wybierz polecenie **Rysuj według zaznaczenia**, a następnie kliknij **Rysuj** i w oknie **Wybór stylu rysowania** ustal szerokość linii, np. na 3 piksele.
7. Usuń zaznaczenie (w menu **Zaznaczenie** kliknij **Brak**) i zapisz plik w formacie XCF, aby mieć dostęp do wszystkich warstw.




**ZAPAMIĘTAJ!**

**Prawo autorskie** chroni prawa twórcy dzieła. Autor decyduje o pierwszym udostępnieniu utworu publiczności i sprawuje nadzór nad sposobem korzystania z utworu. Zanim wykorzystasz utwory innych osób w projektach, które nie mają charakteru ćwiczeniowego, zapoznaj się z licencją. Jeśli **licencja** pozwala na wykorzystanie utworu, oznacz autora oraz dodaj informację o licencji.

### ZMIANA FORMATU W PROGRAMIE GIMP

Podstawowe formaty plików grafiki bitmapowej to:

- **BMP** – umożliwiający zapis bez kompresji lub z kompresją bezstratną, która pozwala na odtworzenie informacji z postaci skompresowanej do postaci pierwotnej;

- **JPG** – umożliwiający zapis z kompresją stratną, co zwykle powoduje zmniejszenie rozmiaru pliku (z tego powodu JPG jest używany w aparatach cyfrowych);
- **GIF** – umożliwiający zapis z kompresją bezstratną i przechowywanie wielu klatek animacji, obsługujący przezroczystość, ograniczający liczbę kolorów do 256;
- **PNG** – umożliwiający zapis z kompresją bezstratną, obsługujący przezroczystość.

Podczas edycji GIMP zapisuje informacje o obrazie bez kompresji w specyficznym dla tego programu formacie XCF. Aby otrzymać plik w formacie JPG, PNG, GIF lub innym, należy obraz wyeksportować i samodzielnie zmienić rozszerzenie w nazwie lub wybrać typ pliku z listy.

**POSZUKAJ W INTERNECIE**

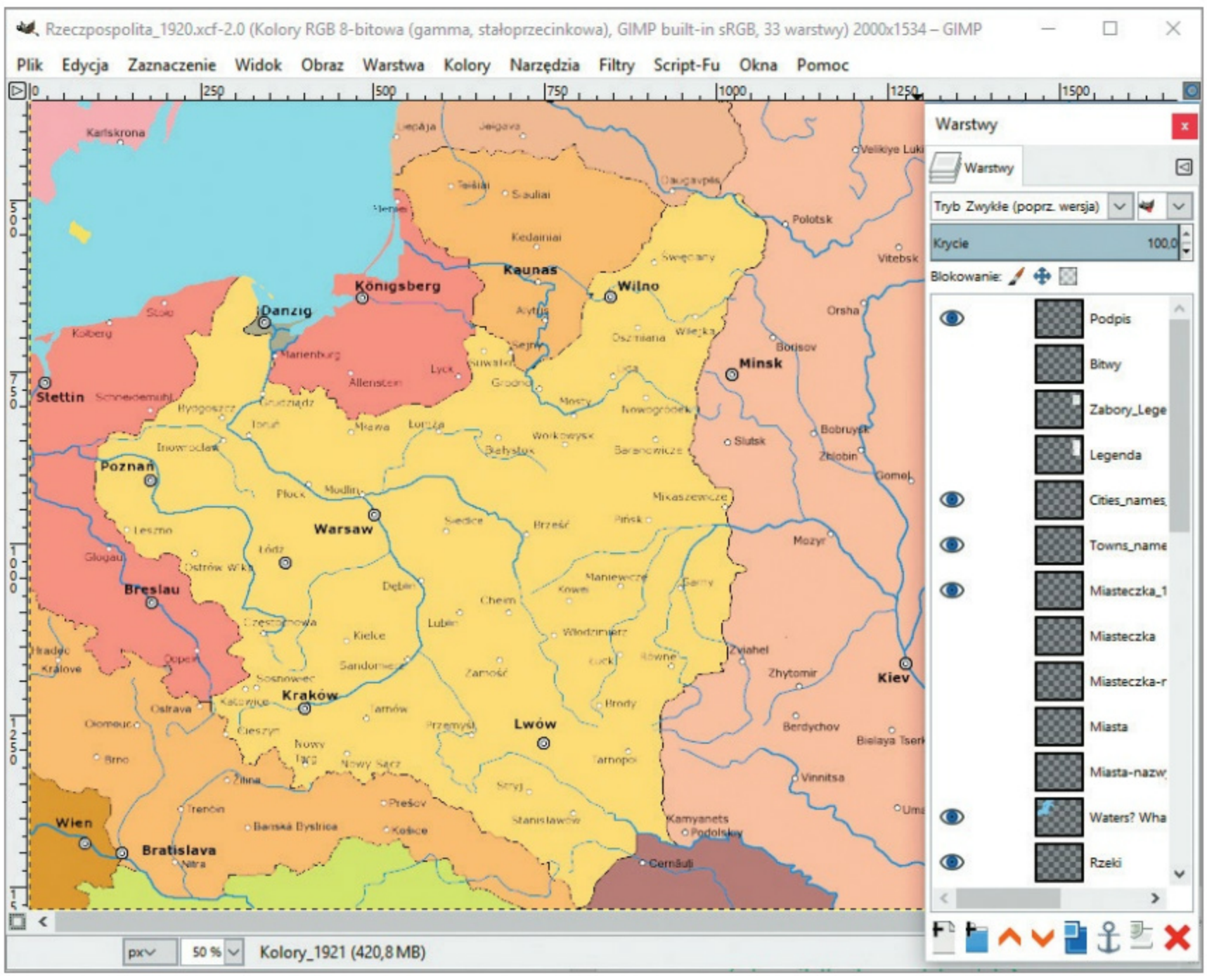
Sprawdź, jakie są zalety formatu WebP, za którego rozwój odpowiada firma Google.

## Ćwiczenie 3. Podgląd warstw gotowego rysunku

Wyszukaj w Wikimedii Commons mapę historyczną z warstwami w formacie XCF, np. mapę Polski z 1920 roku. Pobierz plik, otwórz go w edytorze GIMP, przyjrzyj się, jak są zorganizowane warstwy, a następnie wygeneruj mapę:

a. bez rzek;
b. ukazującą tylko większe miasta;
c. ze wszystkimi wodami w kolorze granatowym.

Zapisz każdy plik w formacie PNG.




1. Kliknij „oko” obok nazwy warstwy Rzeki.

2. Ukryj nazwy większych miast i oznaczające je punkty – kliknij „oko” obok warstw Towns_names_1920 i Miasteczka_1920.

3. Zaznacz warstwę, której nazwa zaczyna się od słowa Waters, w **Przyborniku** zmień kolor na granatowy, wybierz narzędzie **Wypełnienie kubełkiem** i pokoloruj morza oraz jeziora.

4. Zapisz gotowy plik w formacie PNG – wyeksportuj plik, a następnie zmień odpowiednio rozszerzenie w jego nazwie lub wybierz je z listy.

   **Plik → Wyeksportuj jako... → Wybór typu pliku (Według rozszerzenia) → Obraz PNG → Wyeksportuj**

## Ćwiczenie 4. Tworzenie animacji

Wyszukaj w internecie zdjęcie z inscenizacji historycznej odgrywanej w słoneczny dzień, a następnie dodaj efekt deszczu i sepii. Zapisz plik jako animację w formacie GIF.

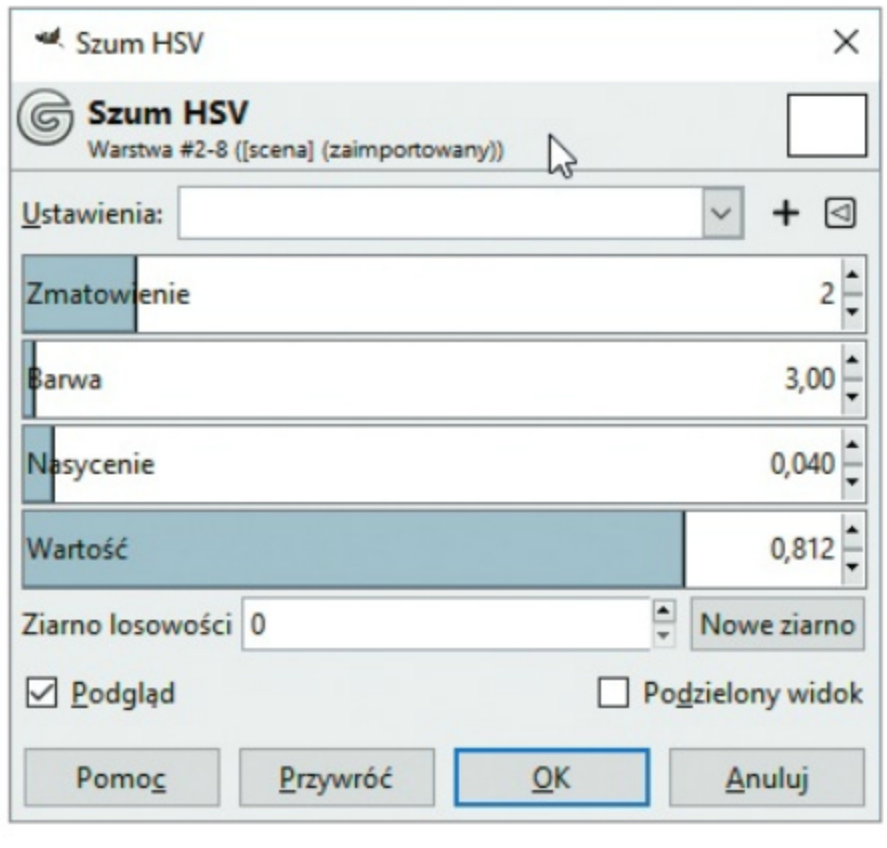




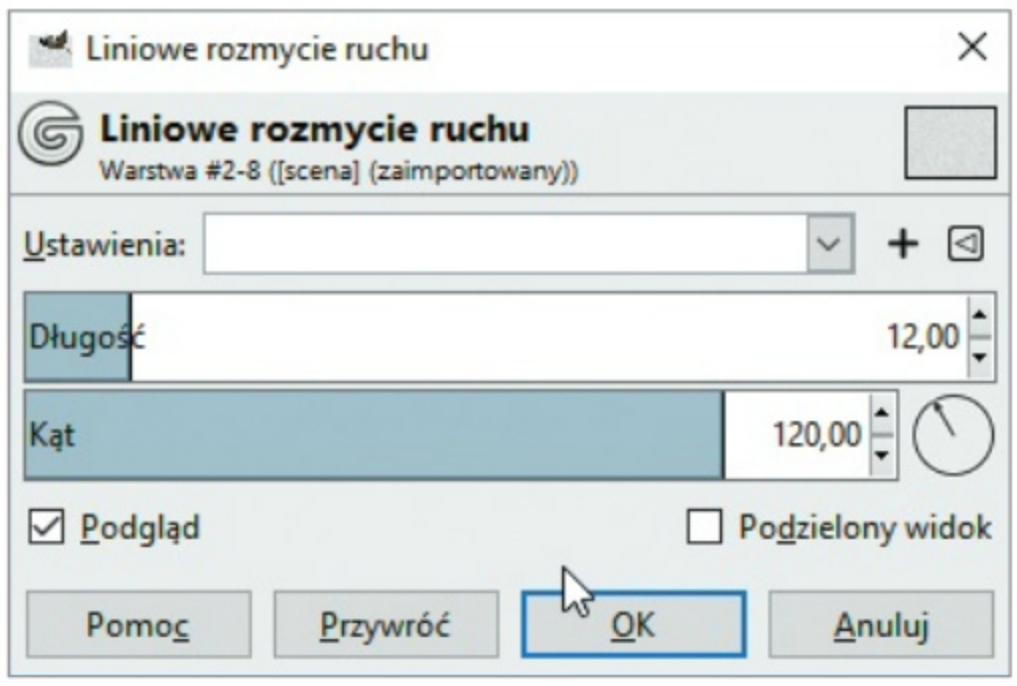
1. Otwórz obrazek i zduplikuj go dwa razy.

2. Nad każdą z warstw dodaj nową warstwę – białą.

3. Na najwyższej białej warstwie wprowadź szum w kolorze tła – wykorzystaj filtr **Szum HSV...** z grupy **Szum**.



4. Wprowadź symulację ruchu – wykorzystaj filtr **Liniowe rozmycie ruchu...** z grupy **Rozmycie**. Aby uzyskać efekt deszczu, ustal długość kreski oraz kąt nachylenia.






5. Powtórz wstawianie efektów **Szum HSV** i **Liniowe rozmycie ruchu** na pozostałych białych warstwach.

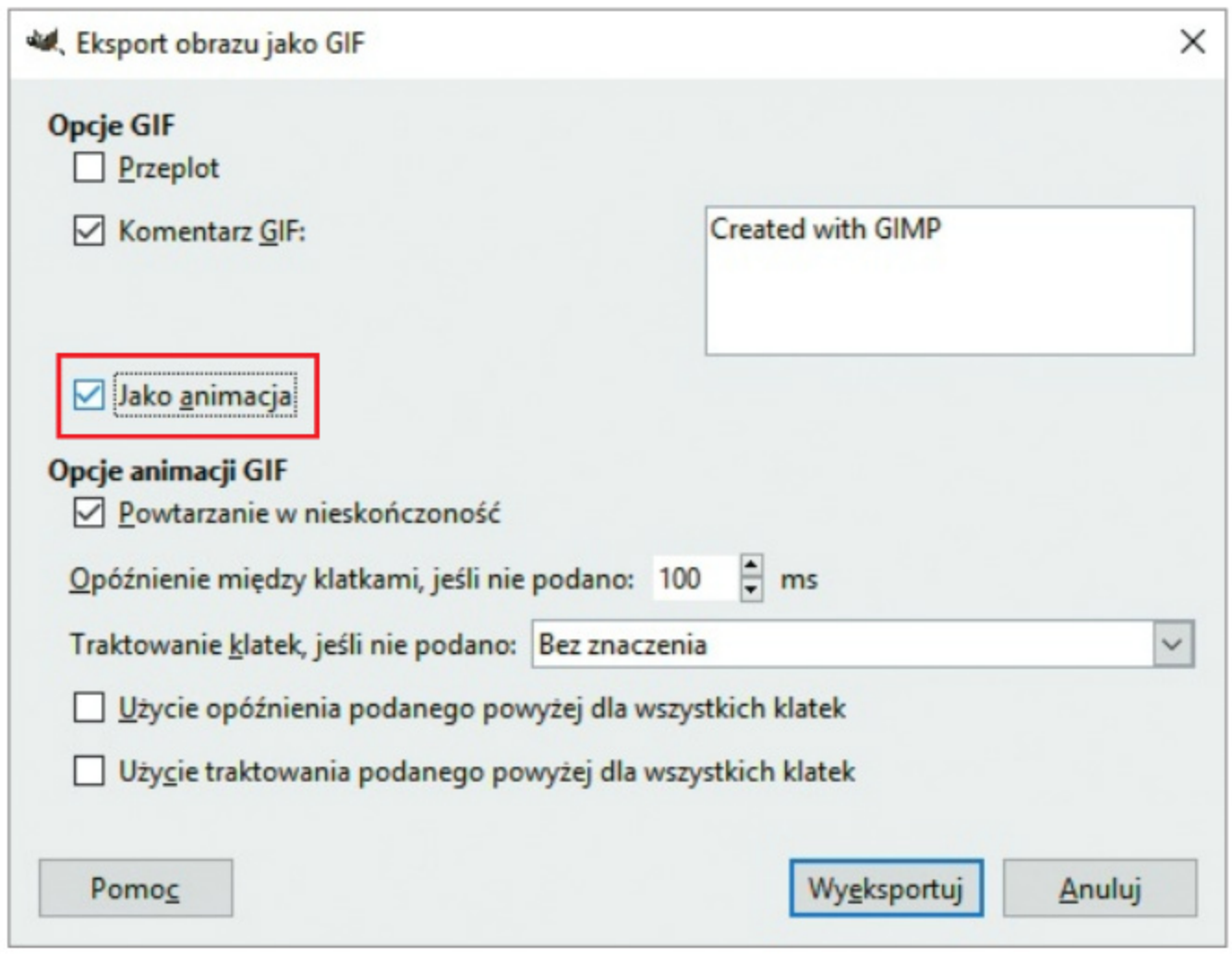
6. Zmień tryb krycia wszystkich warstw z deszczem ze **Zwykłego** na **Mnożenie**, tak aby uzyskać ciemniejszy kolor z łagodniejszymi przejściami.

7. Połącz białe warstwy z deszczem – zaznacz odpowiednie warstwy w oknie **Warstwy** i wybierz z menu podręcznego **Połącz w dół**.

8. Dodaj efekt sepii – nad każdą warstwą kolorową dodaj nową warstwę, pomaluj ją na kolor ciemnożółty, następnie ustaw tryb krycia na **Kolor** i połącz z warstwą kolorową.

9. Zapisz gotową animację.

**Plik → Wyeksportuj jako... → Wybór typu pliku (Według rozszerzenia) → Obraz GIF → Wyeksportuj → Jako animacja → Wyeksportuj**



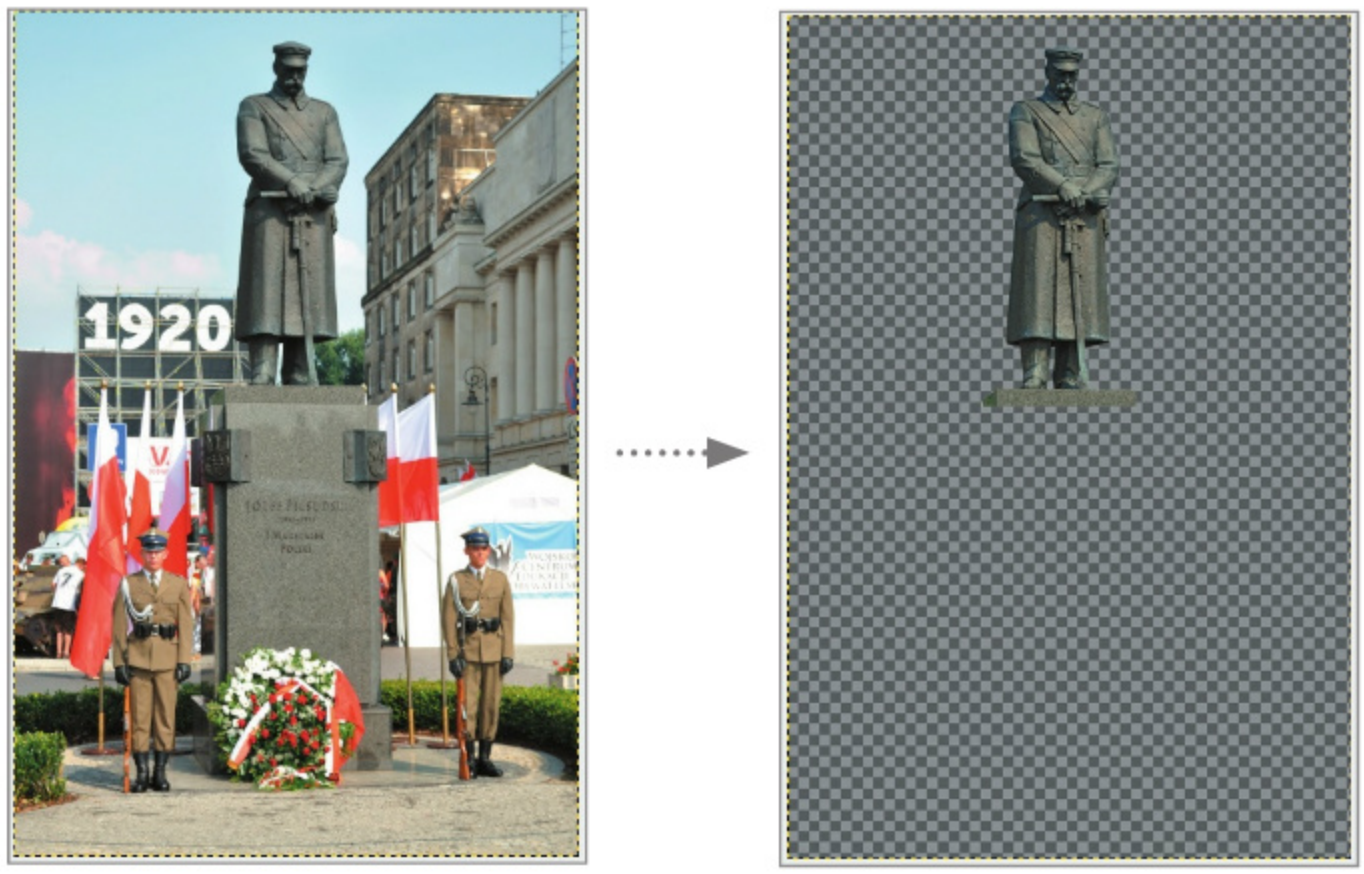

10. Otwórz plik w przeglądarce, aby ocenić uzyskany efekt.

## WYCINANIE

Wycinanie (renderowanie) najczęściej stosuje się podczas tworzenia różnego rodzaju fotomontaży. Wycięty element zdjęcia zapisany na przezroczystej warstwie jest nazywany renderem. Aby wyciąć obiekt z dowolnego zdjęcia, można się posłużyć narzędziem **Zaznaczenie według koloru** lub **Ścieżki**.

### Ćwiczenie 5. Wycinanie elementów

Wytnij fragment z dowolnego zdjęcia i zapisz jako osobny plik na przezroczystym tle.




## PODSUMOWANIE

- Warstwy zapewniają łatwy dostęp do tworzonych na nich elementów grafiki i pozwalają na niezależną edycję fragmentów obrazu.
- Nowy plik ma jedną domyślną warstwę – tło.
- Warstwa położona wyżej w oknie warstw przesłania tę, która jest niżej.
- Edytować można tylko warstwę aktualnie zaznaczoną. Spięte spinaczem warstwy można edytować jednocześnie.
- Za pomocą warstw oraz filtrów można stworzyć animację w formacie GIF.
- Wycięty element zdjęcia zapisany na przezroczystej warstwie to render.

## PYTANIA SPRAWDZAJĄCE

1. W jakim celu stosuje się warstwy?
2. Czy kolejność warstw ma wpływ na wygląd rysunku?
3. Czy wszystkie warstwy muszą być widoczne?

## PYTANIA QUIZOWE

1. Jak brzmi jedna z głównych zasad pracy na warstwach?
   **a.** Rozmiary warstw są takie same.
   **b.** Warstwa położona wyżej przesłania warstwę położoną niżej.
   **c.** Nazwy warstw są oznaczone kolejnymi liczbami.
   **d.** Nie można edytować dwóch warstw jednocześnie.
2. Jakie rozszerzenie ma obraz zapisany z warstwami?
   **a.** PNG
   **b.** BMP
   **c.** GIF
   **d.** XCF
3. Co jest skutkiem kliknięcia ikony w postaci otwartego „oka” obok nazwy warstwy?
   **a.** Ukrycie warstwy.
   **b.** Skasowanie warstwy.
   **c.** Zduplikowanie warstwy.
   **d.** Zgrupowanie warstwy z kolejną warstwą.

## ZADANIA DODATKOWE

### Zadanie 1. Zaproszenie

Przygotuj zaproszenie na inscenizację historyczną z okazji bitwy warszawskiej 1920 r. W projekcie wykorzystaj warstwy.

### Zadanie 2. Kolorowy element

Zapisz kolorowy obrazek w odcieniach szarości z jednym kolorowym elementem. Zastosuj renderowanie.





# 8. Edycja fotografii

- Korygowanie zniekształceń
- Modyfikacja kolorystyczna zdjęcia
- Kompozycja zdjęcia i kadrowanie obrazu

## ZADANIE NA START

### Symulacja aparatu fotograficznego

Poniżej zamieszczono to samo zdjęcie przy różnych wartościach przysłony (f/16 i f/2). Wyszukaj w sieci symulator robienia zdjęć i podaj adres aplikacji. Sprawdź, jakie parametry ekspozycji należy dobrać, by uzyskać małą głębię ostrości (rozmyte tło).





## RETUSZ FOTOGRAFII

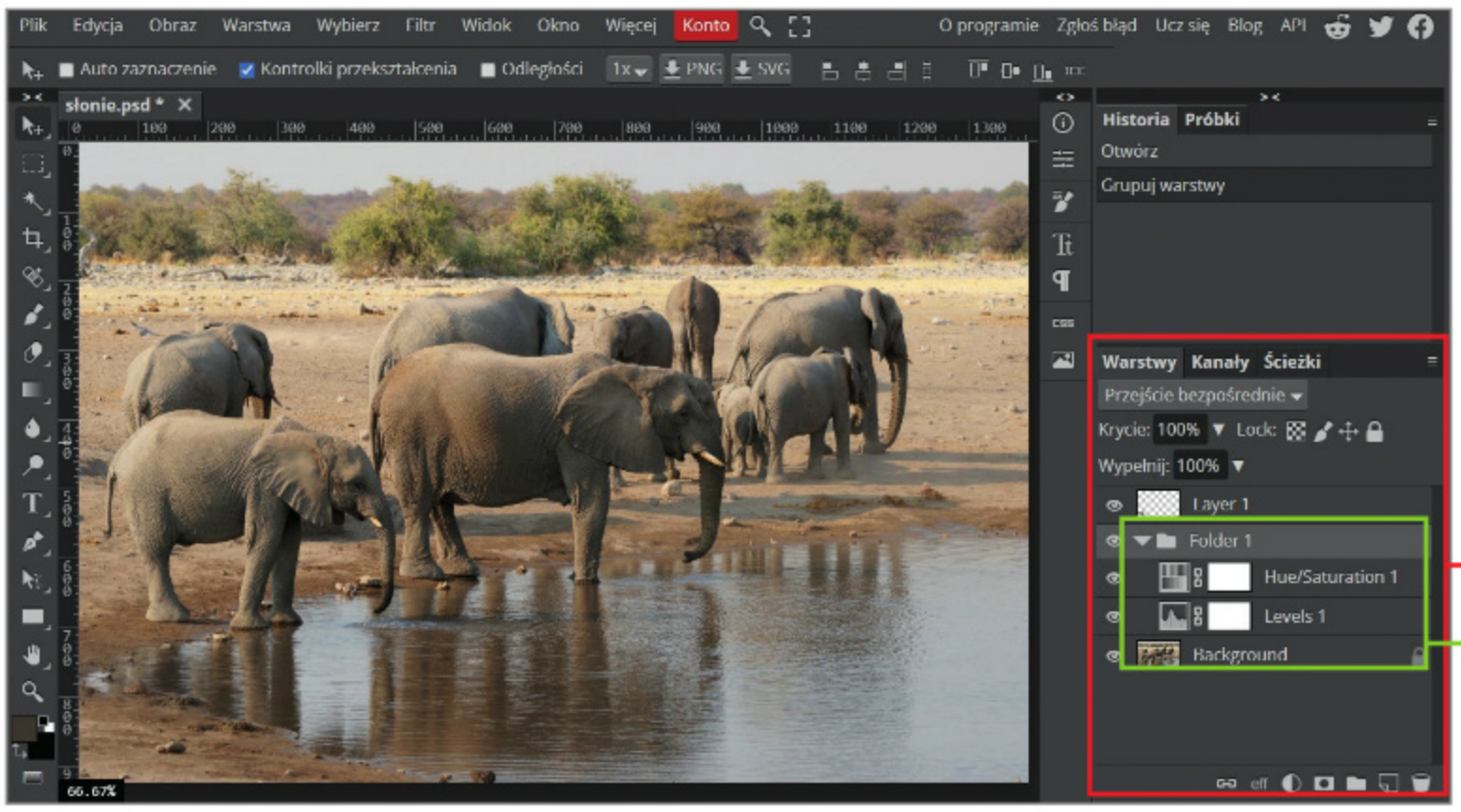
W sztuce fotografowania dobranie odpowiednich parametrów technicznych okazuje się kluczem do osiągnięcia pożądanego efektu. Niestety, zdarza się, że mimo największych starań fotografa efekt nie jest zadowalający i zdjęcie wymaga obróbki cyfrowej. Czasem trzeba dostosować fotografię do przekazu lub artystycznej wizji, ukryć coś lub – przeciwnie – zaakcentować. Niejednokrotnie są to zmiany nieznaczne, wręcz subtelne.

**ZAPAMIĘTAJ!**

**Retusz fotografii** to modyfikacja zdjęcia przez wykadrowanie obrazu, usunięcie usterek i ewentualną zmianę kolorystyki, a także wprowadzanie zmian niedotyczących bezpośrednio kadru, np. dodanie ramki.

GIMP jako klasyczny desktopowy edytor grafiki rastrowej oferuje bogaty zestaw narzędzi, które umożliwiają zaawansowaną obróbkę fotografii, w tym dowolne kadrowanie, zmianę perspektywy, nachylenia, barwy, kontrastu, nasycenia kolorów oraz sporą paletę filtrów i efektów. Podstawową wersję programu można rozbudować, np. za pomocą wtyczki z filtrami Instagram Effects.

Osoby, które preferują pracę online, mogą skorzystać z edytora obrazów rastrowych i wektorowych Photopea (**www.photopea.com**), który bardzo dobrze sprawdza się jako edytor fotografii. Pliki zapisuje się lokalnie na dysku. Warto pamiętać, że wszystkie pliki, w tym PNG lub JPG, są konwertowane na format PSD, dzięki czemu po ponownym otwarciu można je edytować. Po zakończeniu edycji można je wyeksportować do formatu PNG, JPG, SVG, GIF lub PDF.




Warstwy są renderowane jedna na drugiej – w danej chwili można modyfikować tylko jedną warstwę. Pozostałe warstwy mogą być w tym czasie widoczne lub ukryte, w zależności od potrzeb.

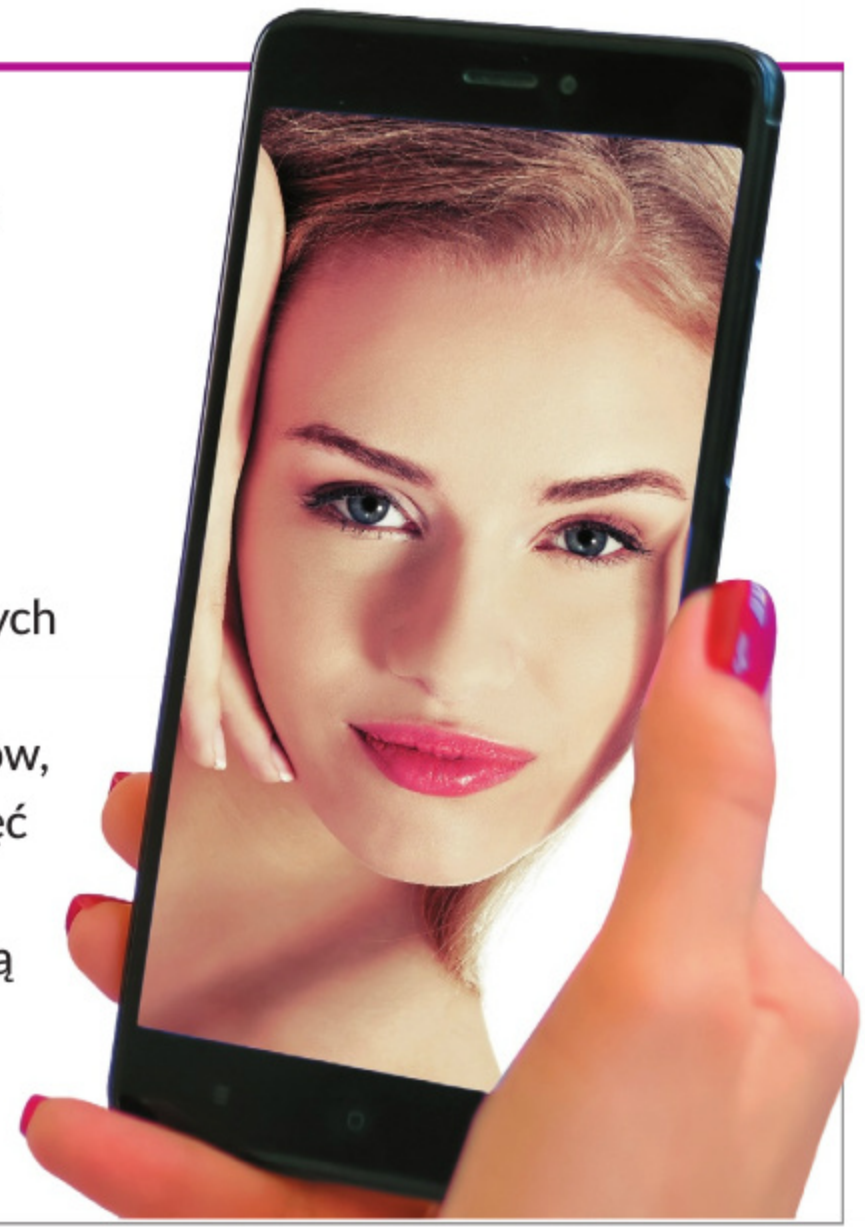
W przypadku większej edycji warto korzystać z opcji tworzenia folderów warstw, która pozwala na uporządkowanie powiązanych warstw.

**Rys. 1.** Zdjęcie w trakcie edycji w programie Photopea z widocznymi 4 warstwami

W przypadku obróbki na smartfonach funkcje dostępne w trybie edycji zdjęć umożliwiają podstawową edycję, m.in. przycięcie, obrót, nakładanie filtrów i innych efektów (np. naklejek czy nakładek typu makijaż albo mozaika), modyfikację parametrów typu jasność, kontrast, nasycenie, ostrość. Można też pobrać aplikacje stworzone specjalnie do edycji zdjęć, np. Snapseed lub Photo Lab Picture Editor.

**WIEM WIĘCEJ**

Użytkownicy portali społecznościowych często tak mocno retuszują swoje zdjęcia, że te niewiele mają wspólnego z rzeczywistością. Ponieważ kreowanie nierealistycznych wizerunków jako ideałów piękna (m.in. zbyt wąskiej talii, nadmiernie uwydatnionych mięśni czy nawet przesadnie powiększonych ust) może mieć negatywny wpływ na samoocenę młodych internautów, niektóre serwisy społecznościowe, np. Instagram, ograniczyły liczbę oferowanych filtrów, a niektóre kraje wprowadziły nakaz oznaczania zdjęć retuszowanych. Przykładowo: w Norwegii użytkownicy Snapchata, Instagrama i TikToka muszą oznaczać retuszowane zdjęcia wykorzystywane w postach sponsorowanych czy zawierających reklamę – pod groźbą kary pieniężnej.



## KORYGOWANIE ZNIEKSZTAŁCEŃ

Przekrzywienie obrazu, niewypoziomowana linia horyzontu albo „walące się" budynki to częsty problem związany z niedostatecznym wypoziomowaniem aparatu, przyjęciem nieodpowiedniej perspektywy albo stosowaniem obiektywów szerokokątnych. W GIMP-ie tego typu defekty fotografii można poprawić za pomocą narzędzi z grupy **Narzędzia przekształcania**.

„Prostowanie" budynków staje się możliwe dzięki narzędziu **Perspektywa**. W opcjach tej funkcji, wyświetlanych po kliknięciu właściwej ikony, można określić kierunek procesu, interpolację oraz przycinanie. Przesunięcie punktu w centrum obrazu powoduje zmianę umiejscowienia środka perspektywy.



**Rys. 2.** Zdjęcie oryginalne i po zmianie perspektywy

Narzędzie **Obrót** pozwala obrócić aktywną warstwę, zaznaczenie lub ścieżkę pod wybranym kątem. Zastosowanie tego narzędzia wiąże się z lekkim zmniejszeniem zdjęcia, ponieważ po wykonaniu operacji obraz należy odpowiednio przyciąć.



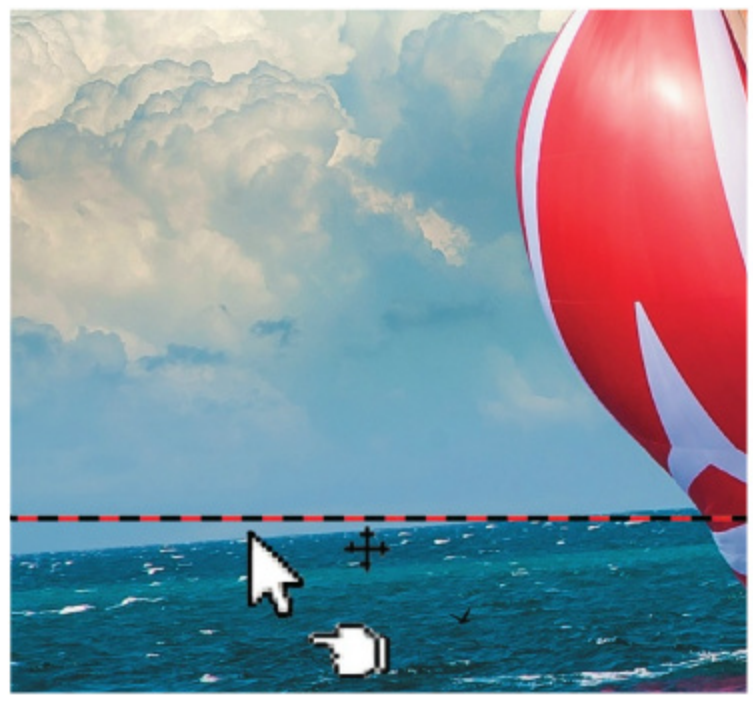
**Rys. 3.** Zdjęcie oryginalne i po wykonaniu nieznacznego obrotu w prawo

Po wykonaniu obrotu nierzadko trzeba zmienić albo zmodyfikować proporcje obrazu. Należy sięgnąć wówczas po narzędzie **Skalowanie** i odpowiednio rozciągnąć zdjęcie, ewentualnie skorzystać z **Kadrowania** (grupa **Narzędzia przekształcania**) albo z opcji **Kadruj do zawartości** bądź **Dokładnie przytnij** (umieszczonych w grupie **Obraz**).

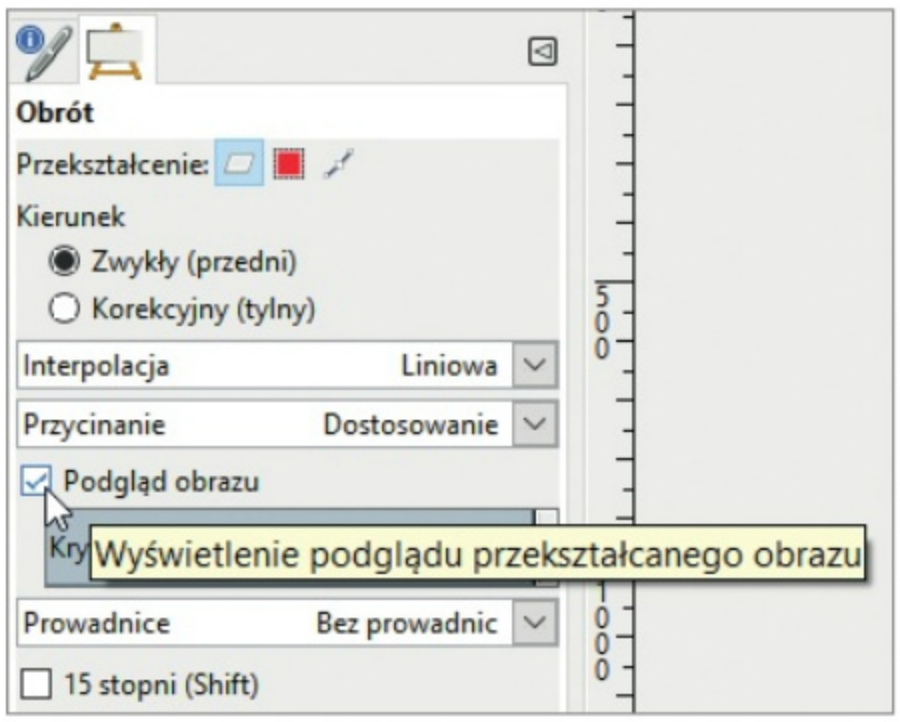
## Ćwiczenie 1. Poprawianie orientacji zdjęcia

Przeanalizuj poniższy proces edycji zdjęcia z rys. 3, a następnie wybierz trzy zdjęcia, które wymagają skorygowania zniekształceń, i odpowiednio je wyretuszuj. Zastosuj – według potrzeby – operacje takie, jak: obrót, przesunięcie, skalowanie, perspektywa oraz kadrowanie z zachowaniem proporcji.

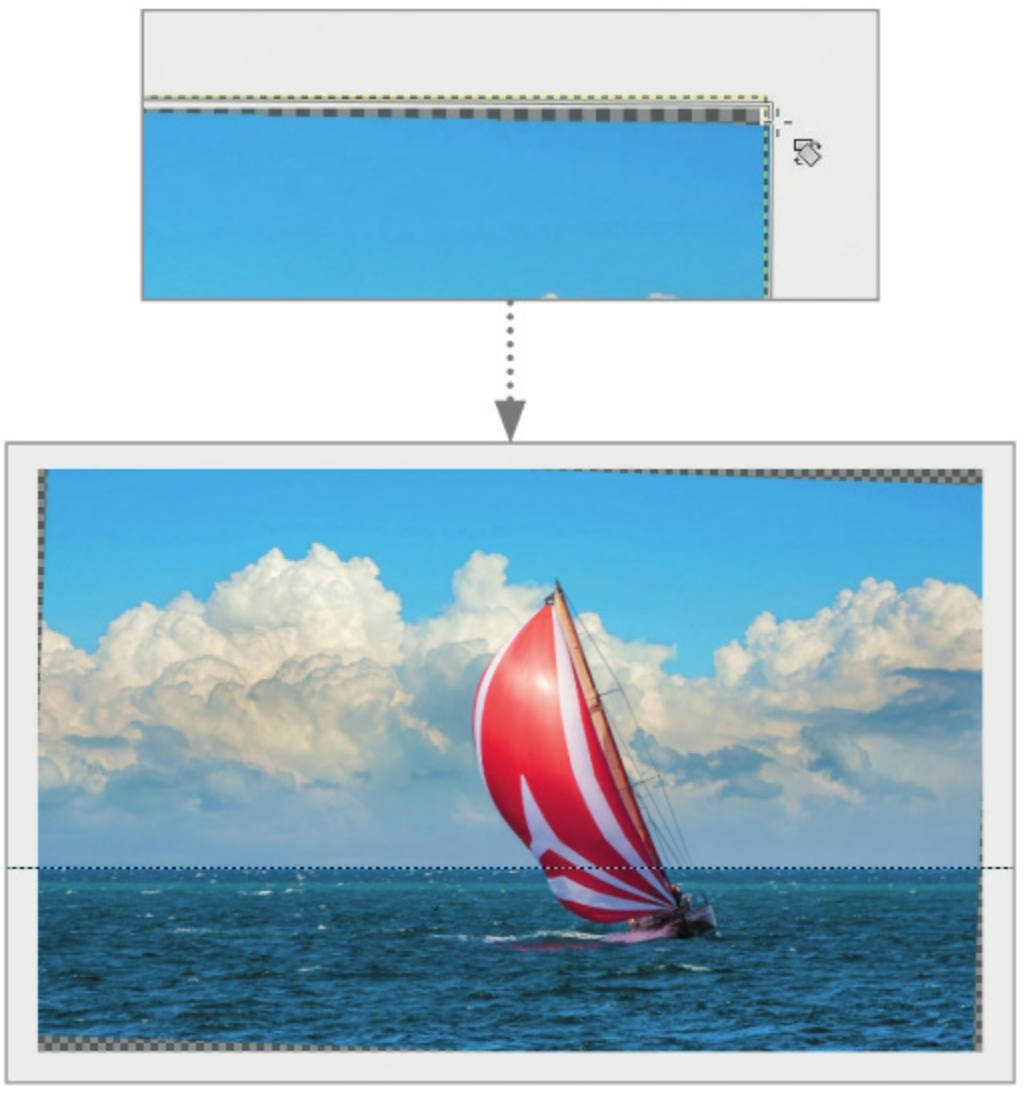
1. Przeanalizuj zdjęcie i wypunktuj elementy wymagające korekty. W tym przypadku konieczne jest wypoziomowanie linii horyzontu.
2. Kliknij w przyborniku narzędzie **Obrót**.
3. Wstaw prowadnicę poziomą, tak aby łatwiej było kontrolować linię horyzontu – kliknij i przytrzymaj wciśnięty lewy przycisk myszy na podziałce, po czym przesuń kursor w dół na linię horyzontu. Gdy ustawisz prowadnicę w pożądanym miejscu, zwolnij przycisk myszy.



4. Zanim przystąpisz do korygowania zniekształcenia, upewnij się, że w opcjach narzędzia **Obrót** zostało zaznaczone pole wyboru przy **Podglądzie obrazu**.




5. Chwyć za róg zdjęcia i obróć je tak, aby linia morza była poziomo. Kieruj się ustawieniem prowadnicy. Na koniec w oknie dialogowym **Obrót** zatwierdź zmiany przyciskiem **Obróć**.



6. Popraw proporcje – kliknij w przyborniku narzędzie **Skalowanie** i odpowiednio rozciągnij obraz tak, aby wypełniał całą warstwę.

7. Gotowy obraz wyeksportuj do pliku JPG.

## KORYGOWANIE KOMPOZYCJI

Każda fotografia powinna być właściwie skomponowana. Aby ustalić kadr, warto kierować się zasadą trójpodziału. Obraz widoczny w wizjerze aparatu lub na ekranie smartfona należy podzielić pionowo i poziomo na trzy równe części. Miejsca, w których krzyżują się linie, to tzw. mocne punkty obrazu – właśnie tam powinno znaleźć się coś istotnego. Właściwe wyeksponowanie elementów kompozycji wpływa na spójność i harmonię zdjęcia oraz dodaje mu dynamiki.




**Rys. 4.** Podczas ustalania kadru warto włączyć w aparacie linie siatki

W przypadku fotografii krajobrazu zwykle lepiej nie umieszczać linii horyzontu na środku kadru. Układ niesymetryczny, gdy linia horyzontu nie dzieli obrazu na połowy, jest bardziej interesujący.

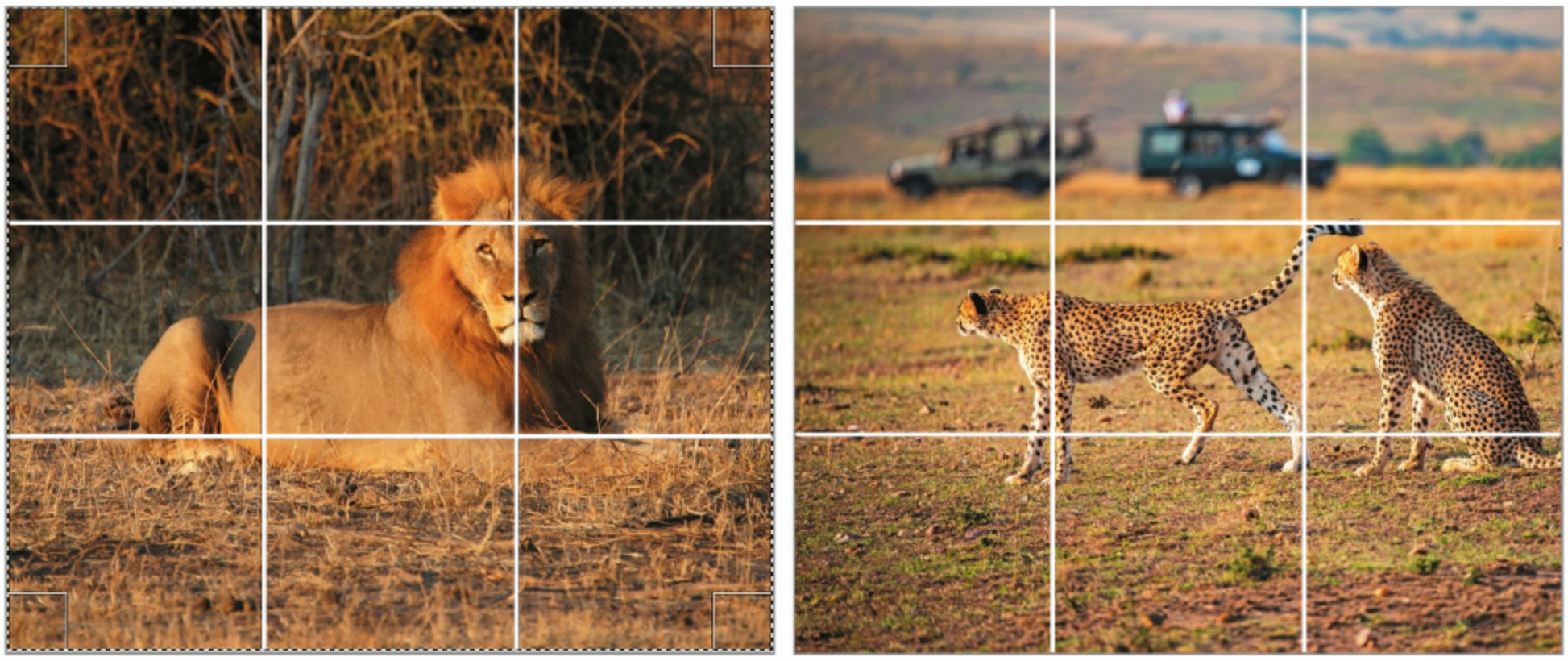


**Rys. 5.** Przykłady umieszczenia linii horyzontu w kadrze

**W PRAKTYCE**

Należy pamiętać, że **układ automatycznego ustawiania ostrości (autofokus)** ustawia punkty AF w centrum ekranu aparatu. Zamiast odruchowo koncentrować się na tym obszarze, warto poświęcić chwilę na odpowiednie ustawienie kadru i skupić ostrość na głównym obiekcie fotografowanego obrazu. W przypadku smartfona wystarczy palcem dotknąć tego obiektu na ekranie – pojawią się wówczas ramka otaczająca obszar ustawiania ostrości oraz suwak MF umożliwiający manualne operowanie ostrością zdjęcia.

Aby poprawić kompozycję zdjęcia, trzeba je odpowiednio przyciąć. W GIMP-ie tego typu operację umożliwia **Kadrowanie** z grupy **Narzędzia przekształcania**. W opcjach można określić sposób kadrowania (np. stałe pod względem proporcji czy rozmiaru) oraz wybrać układ prowadnic pomagający zgrabnie skomponować kadr.



**Rys. 6.** Prowadnice ustawione zgodnie z regułą trójpodziału kadru w edytorze GIMP

**WIEM WIĘCEJ**

Ponieważ wzrok osoby oglądającej zdjęcie automatycznie trafia tam, gdzie zaczyna czytać tekst, dla osób czytających od lewej do prawej najmocniejszy jest lewy górny róg fotografii, natomiast dla osób czytających od prawej do lewej, czyli np. w kulturach arabskiej i hebrajskiej, najmocniejszy jest prawy górny róg. Kompozycja centralna, choć poprawna, sprawdza się tylko w przypadku symetrycznych obrazów. Dlatego należy z rozwagą stosować ją w przypadku zdjęć portretowych, krajobrazowych czy wakacyjnych.

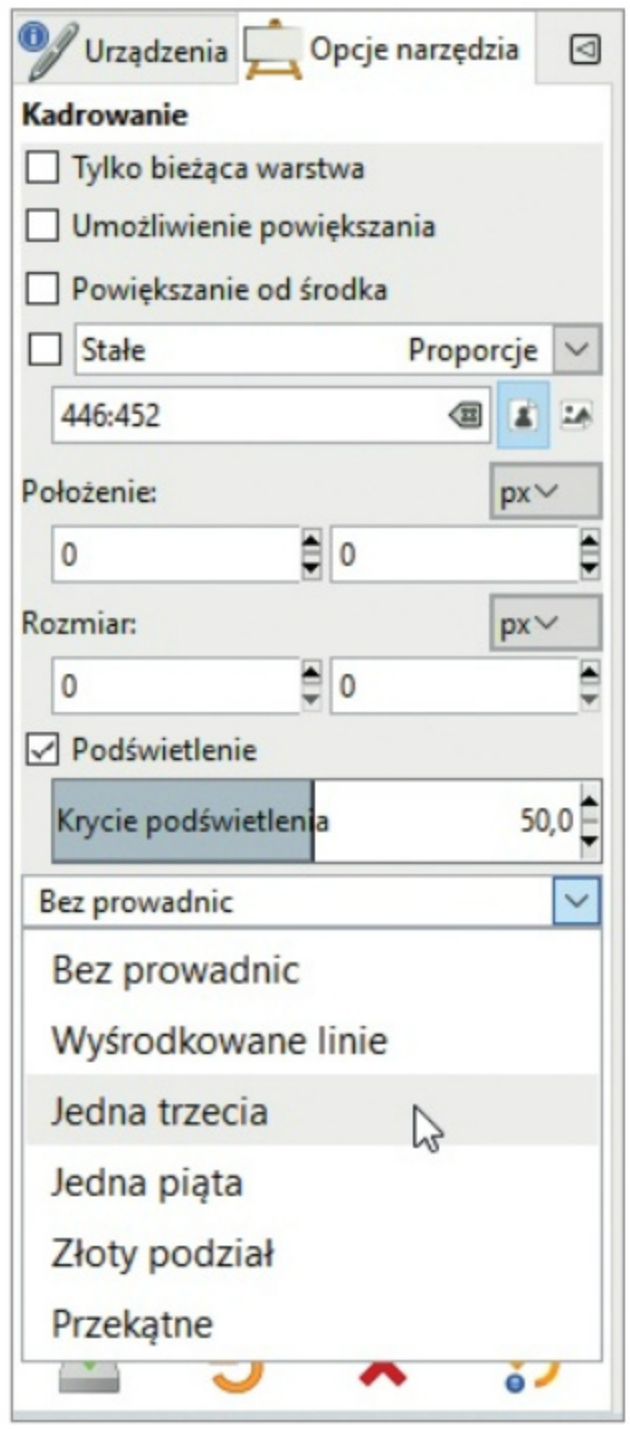
## Ćwiczenie 2. Poprawianie kompozycji

Przeanalizuj zaprezentowany poniżej proces edycji zdjęcia, a następnie wybierz zdjęcie, które wymaga skorygowania kompozycji, i wykadruj je z zachowaniem proporcji. Porównaj oba zdjęcia: oryginalne i zmodyfikowane.

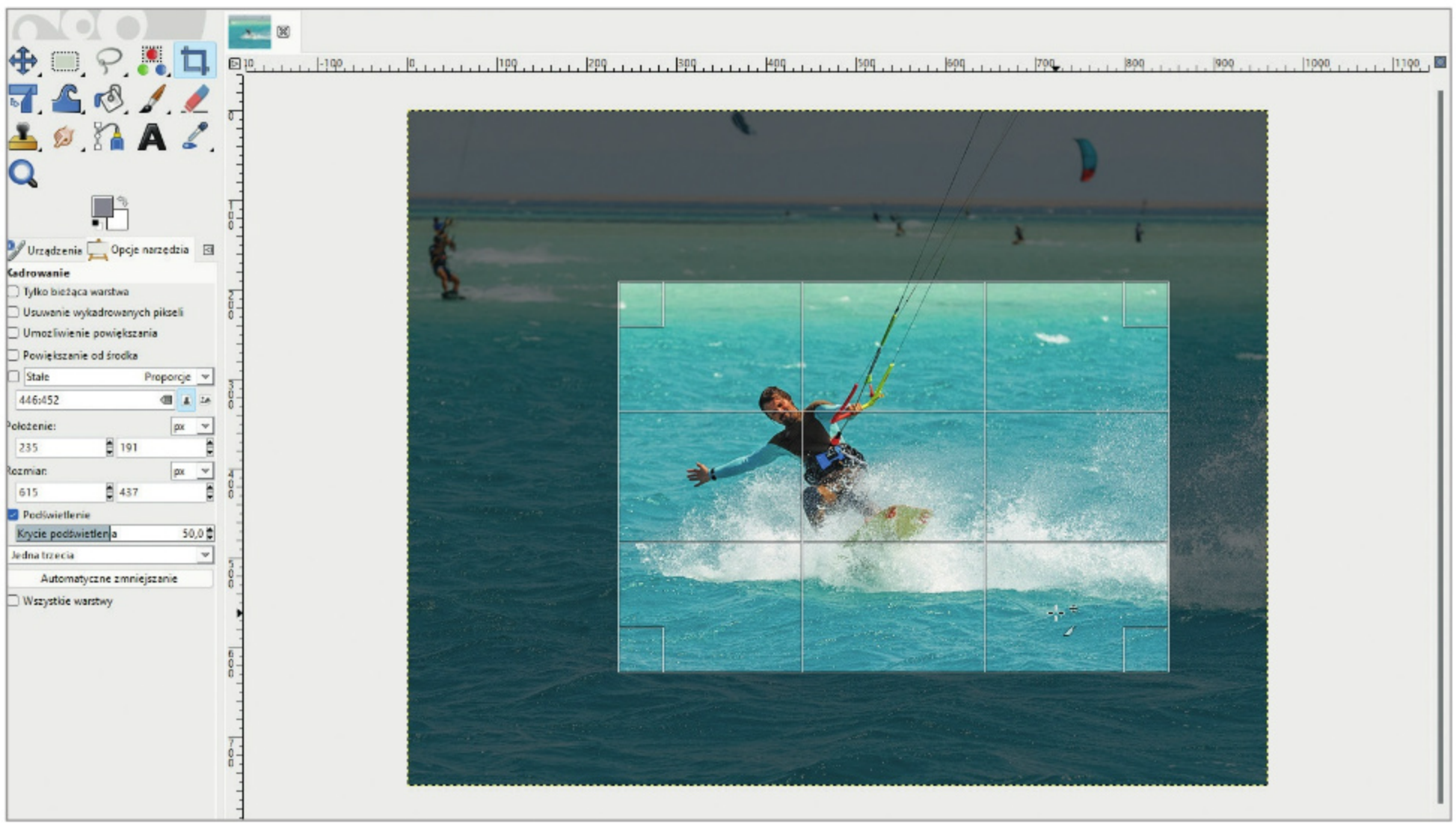
1. Przyjrzyj się zdjęciu i wskaż elementy, które warto wyeksponować. Jeżeli to portret (a więc również selfie), kieruj się zasadą, żeby spojrzenie wyznaczyło kierunek kadrowania. Jeśli osoba fotografowana nie patrzy w obiektyw, należy zostawić więcej miejsca po tej stronie, w którą jest skierowany jej wzrok. Ta sama reguła dotyczy obiektów będących w ruchu.



2. Kliknij w przyborniku narzędzie **Kadrowanie**. W opcjach narzędzia wybierz z listy **Jedna trzecia**, a następnie kliknij w lewym górnym rogu zdjęcia i nie zwalniając lewego przycisku myszy, przeciągnij nią w kierunku prawego dolnego rogu zdjęcia, aby nałożyć na obraz siatkę trójpodziału kadru.




3. W mocnym punkcie z lewej strony umieść to, co chcesz wyróżnić na zdjęciu – w tym przypadku kitesurfera.



4. Wykadruj zaznaczony fragment zdjęcia – naciśnij klawisz **Enter**.




5. Gotowy obraz wyeksportuj do pliku JPG.

## ZMIANA KOLORÓW

Aby skorygować barwy czy jasność, w GIMP-ie wystarczy wybrać opcję **Automatycznie** w menu **Kolory** i zaznaczyć opcje wymienione na liście (np. **Balans bieli** automatycznie poprawia odcień, a **Uwydatnianie koloru** – nasycenie barw). Odpowiednie parametry można też zmodyfikować samodzielnie.

Jeśli zdjęcie ma za mały kontrast, należy skorzystać z opcji **Jasność i kontrast...**, która zgodnie ze swoją nazwą modyfikuje jasność i kontrast obrazu. Aby osiągnąć pożądany efekt, wystarczy odpowiednio przesunąć suwak.

Aby zmienić odcienie lub barwy, warto sięgnąć po **Balans kolorów…** – w oknie, które pojawi się po kliknięciu tej opcji, można określić zakres jasności pikseli, a suwakami nadać im odpowiedni odcień. Opcja **Zachowanie luminacji** pozwala zachować jasność na danym poziomie.

Z kolei opcja **Poziomy…** umożliwia m.in. zmianę naświetlenia zdjęcia przez modyfikację poziomów wejściowych i wyjściowych. Można też za jej pomocą manipulować konkretnym kanałem koloru RGB oraz poprawić balans czerni lub bieli, rozjaśnić lub przyciemnić światło czy poprawić jasność obrazu z zachowaniem kontrastu.



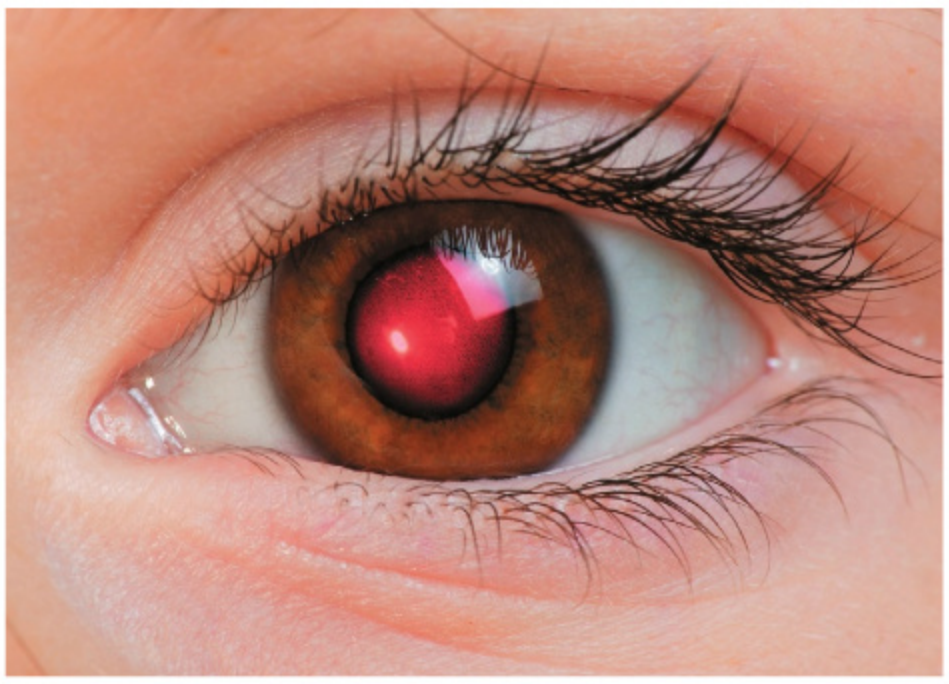
**Rys. 7.** Zdjęcie oryginalne (z lewej) i po zmianie poziomu kolorów

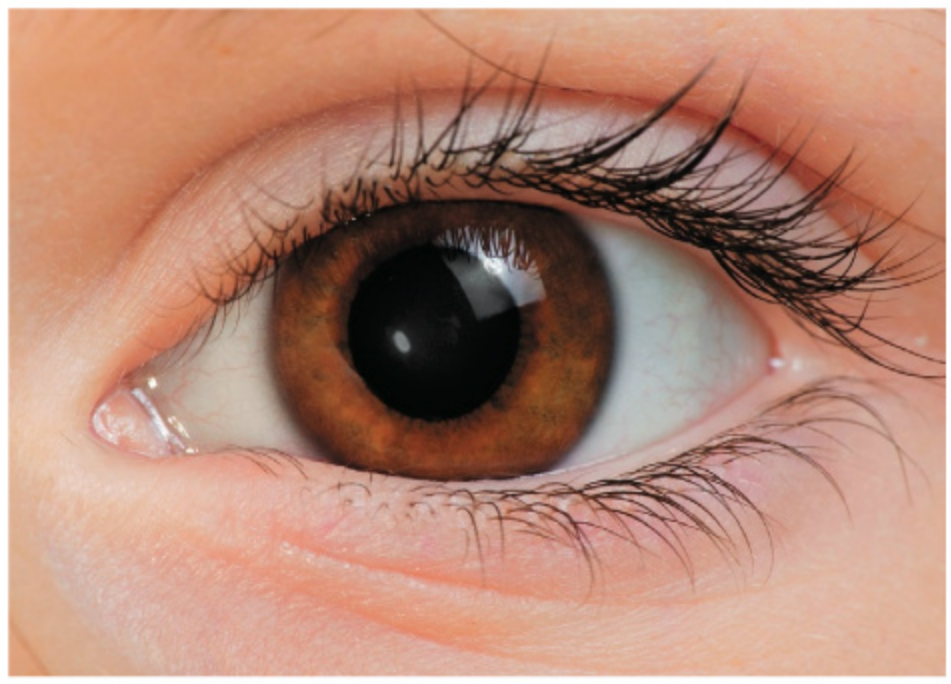
Modyfikację nasycenia, jasności oraz odcienia całości obrazu lub niektórych barw umożliwia opcja **Barwa i nasycenie…** – za jej pomocą można sprawić, że kolory będą bardziej soczyste albo stonowane.

## Ćwiczenie 3. Retusz czerwonych oczu

Gdy osoba jest fotografowana z użyciem lampy błyskowej, patrzy w stronę aparatu i ma rozszerzone źrenice, światło lampy błyskowej odbija się od dna oka i wraca do aparatu fotograficznego, co skutkuje tzw. efektem czerwonych oczu (charakterystyczna czerwona barwa powstaje pod wpływem rodopsyny, występującej w siatkówce). Znajdź zdjęcie z efektem czerwonych oczu i dokonaj jego retuszu w edytorze GIMP. Zastosuj jedną z poniższych metod:

- filtr **Usuwanie efektu czerwonych oczu…** z grupy **Uwydatnianie**,
- opcję **Desaturacja…** z menu **Kolory**,
- opcję **Balans kolorów** z menu **Kolory**,
- zwykłą zmianę kolorystyki.





### WIEM WIĘCEJ

Aby uniknąć efektu czerwonych oczu lub efektu świecących oczu w przypadku zwierząt (błona odblaskowa w oczach zwierząt daje na fotografii kolor niebieski, zielononiebieski lub żółtozielony) w aparatach z wbudowaną lampą błyskową, należy włączyć taki tryb pracy lampy, w którym przed otwarciem migawki i właściwym błyskiem następuje kilka krótkich błysków przystosowujących oko do jasnego światła.

## Ćwiczenie 4. Modyfikacja kolorów

Wybierz zdjęcie z zachodem słońca, a następnie przeanalizuj histogram obrazu, który uwzględnia poziomy pikseli o danej jasności, oraz sprawdź balans bieli. Zmodyfikuj kolorystykę zdjęcia i zapisz kilka różnych jego wersji.

## DODAWANIE I USUWANIE ELEMENTÓW

Zdarza się, że na zdjęciu trzeba dodać jakiś element lub usunąć z niego coś niepożądanego, np. przypadkową osobę w kadrze, przejeżdżający samochód, linię wysokiego napięcia. W edytorze GIMP do tego celu można użyć działających na zasadzie stempla narzędzi **Klonowanie** lub **Łatka** z grupy **Narzędzia rysowania**. Za ich pomocą we wskazanym miejscu umieszcza się piksele skopiowane z innego obszaru. Należy zauważyć, że choć opisane narzędzia mają wspólne opcje, to efekty ich użycia są różne. **Klonowanie** pozwala zastosować wybrany deseń, nieco delikatniejsza **Łatka** dopasowuje się do okolicy.



**Rys. 8.** Zdjęcie oryginalne i po zastosowaniu klonowania do usunięcia samolotu oraz usunięcia i dodania ptaków

W obu przypadkach należy wybierać miękką końcówkę pędzla, aby nakładane piksele lepiej wtapiały się w tło, oraz maksymalne krycie, by szybciej wykonać zadanie. Aby pobrać próbkę pikseli, które najlepiej przysłonią element, należy kliknąć tło obok obiektu do usunięcia z wciśniętym klawiszem **Ctrl**. Trzeba przy tym wziąć pod uwagę kolor, jasność i fakturę, aby efekt był jak najbardziej naturalny (klonowanie można zastosować również w przypadku usuwania drobnych krostek, znamion, przebarwień cery czy piegów – należy tylko zadbać o to, by zaznaczyć odpowiednią tonację skóry, a dla większej precyzji lepiej skorzystać z wtyczki **Wavelet Decompose**). Warto pracować na dużym powiększeniu, ale od czasu do czasu oddalić podgląd, aby ocenić efekt z perspektywy.

W przypadku architektonicznych ujęć z widocznie zniekształconą perspektywą, np. płyt chodnikowych lub posadzki o równomiernym wzorze albo wieżowca, z którego szklanej ściany trzeba usunąć reklamę, warto sięgnąć po **Klon perspektywy**. Skopiowane piksele umieszczane są wówczas zgodnie z perspektywą.

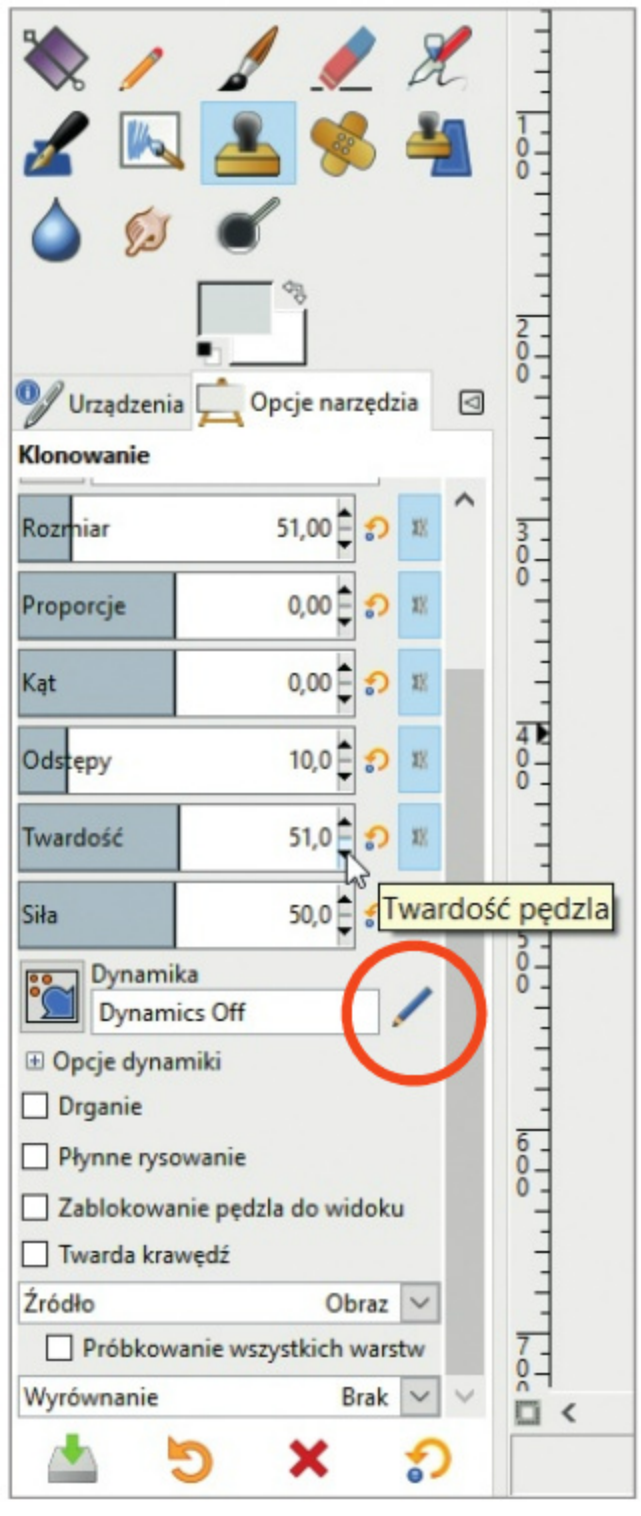
## Ćwiczenie 5. Usuwanie elementów zaburzających ujęcie

Przeanalizuj zaprezentowany poniżej proces edycji zdjęcia, a następnie stosując klonowanie, usuń z wybranego przez siebie zdjęcia obiekt zaburzający kadr.



1. Przeanalizuj zdjęcie i wskaż element zakłócający kompozycję, którego nie da się wykadrować. W tym przypadku są to pływający ludzie.

2. W przyborniku kliknij narzędzie **Klonowanie**, a następnie w opcjach narzędzia odpowiednio zmodyfikuj rozmiar i twardość pędzla.



3. Wskaż myszą fragment, który chcesz zduplikować, a następnie naciśnij klawisz **Ctrl** i lewy przycisk myszy.

4. Ustaw kursor w nowym położeniu i zwolnij przycisk.

5. Powtarzaj operację tak długo, aż usuniesz elementy zaburzające ujęcie.



## DODAWANIE EFEKTÓW

GIMP oferuje sporo narzędzi, za pomocą których można uatrakcyjnić edytowane zdjęcie przez dodanie ciekawego efektu.

Zdjęcie można łatwo zamienić w grafikę: **Barwienie…** i **Koloryzuj…** z menu **Kolory** sprawiają, że obraz zostaje jednolicie zabarwiony na dany kolor i całe zdjęcie zyskuje np. odcień fioletowy; **Redukcja kolorów...** zmniejsza liczbę kolorów obrazu, a **Progowanie...** ogranicza kolory do dwóch – czarnego i białego.



**Rys. 9.** Dodawanie efektów

Aby miejscowo – za pomocą pędzla – rozmyć lub wyostrzyć konkretny element obrazu, można zastosować **Rozmywanie/wyostrzanie** z grupy **Narzędzia rysowania**. Pod wpływem pociągnięć pędzla piksele na danym obszarze odpowiednio zleją się z sąsiednimi lub zyskają większą ostrość (efekt będzie tym mocniejszy, im więcej wykona się pociągnięć na danym obszarze). W przypadku większego fragmentu warstwy lub całej warstwy lepiej sięgnąć po filtry. Wspomniane w lekcji 6 **Rozsmarowanie** pozwala utworzyć smugi.

Ciekawy efekt zmniejszonej głębi ostrości, gdy jeden element zdjęcia jest wyróżniony, a tło nieostre, bądź część zdjęcia jest ostra, a część nie, można uzyskać dzięki filtrowi **Rozmycie Gaussa** z grupy filtrów **Rozmycie**.



**Rys. 10.** Zdjęcie oryginalne i po zastosowaniu rozmycia

## Ćwiczenie 6. Wyróżnianie fragmentów obrazu

Przekształć wybrane zdjęcie tak, aby uzyskać efekt zmniejszonej głębi ostrości fragmentu obrazu.

## PODSUMOWANIE

- Zanim opublikuje się fotografię lub włączy ją do prezentacji, często trzeba poprawić estetykę kompozycji (np. poprzez wykadrowanie obrazu albo zbalansowanie kolorystyki), dostosować wygląd do przekazu lub artystycznej wizji.
- Edytory grafiki offline (np. GIMP) i online (np. Photopea) oferują zestaw narzędzi, za pomocą których można m.in. wyprostować linię horyzontu, poprawić nasycenie kolorów, skorygować kadr czy usunąć zbędne obiekty zaburzające ujęcie.
- Narzędzia **Perspektywa** i **Obrót** z grupy **Narzędzia przekształcania** pozwalają skorygować zniekształcenia typu przekrzywienie obrazu, niewypoziomowana linia horyzontu albo „walące się" budynki. Narzędzie **Kadrowanie** z tej samej grupy umożliwia korektę kompozycji i odpowiednie wyeksponowanie kluczowych elementów obrazu. Narzędzia z menu **Kolor** pozwalają na retusz niedoświetlenia lub prześwietlenia, a także zniekształconych lub zafałszowanych kolorów.

## PYTANIA SPRAWDZAJĄCE

1. Jakiego typu przekształcenia mogą poprawić jakość obrazu?
2. Na czym polega korekta kolorystyczna zdjęcia i w jaki sposób można ją wykonać?
3. W jaki sposób można poprawić kompozycję zdjęcia?

## PYTANIA QUIZOWE

1. Co staje się możliwe dzięki narzędziu **Perspektywa** w edytorze GIMP?
   **a.** Usunięcie dużych obiektów z pierwszego planu.
   **b.** Usunięcie obiektów z tła.
   **c.** Prostowanie pochyłych ścian na zdjęciu.
   **d.** Dodanie trzeciego wymiaru do fotografii.
2. Jak należy podzielić kadr zgodnie z zasadą trójpodziału?
   **a.** Pionowo i poziomo na trzy równe części, tak by w przecięciach (mocnych punktach) umieścić najważniejsze obiekty.
   **b.** Na trzy strefy: najbliższą, środkową i najdalszą, by uzyskać efekt głębi.
   **c.** Na trzy pionowe obszary – lewy, środkowy i prawy, co zwiększa dynamikę zdjęcia.
   **d.** Na trzy strefy kolorystyczne, co pozwala zbalansować kolorystykę zdjęcia.
3. Na czym polega progowanie?
   **a.** Na otrzymaniu obrazu binarnego.
   **b.** Na wyostrzaniu elementów obrazu.
   **c.** Na uwydatnianiu wybranego koloru.
   **d.** Na zmniejszaniu głębi ostrości.

## ZADANIE DODATKOWE

### Zadanie 1. Pocztówka z podróży

Wyszukaj kilka zdjęć charakterystycznych dla danego miejsca. Ułóż z nich kolaż w formacie pocztówki i dopasuj kompozycję. Dodaj napis, a w razie potrzeby wykadruj zdjęcia, zmień kolorystykę oraz wykonaj inne zabiegi.



# SPOSÓB NA DOBRE SELFIE

**Szacuje się, że na całym świecie codziennie wykonuje się ponad milion selfie. Aby twój smartfonowy autoportret dobrze się prezentował, wystarczy przestrzegać kilku zasad.**



## OŚWIETLENIE

- Im mniej światła, tym gorsza jakość zdjęcia.
- W pomieszczeniu podejdź jak najbliżej naturalnego źródła światła i ustaw się tak, by padało na twarz równomiernie.
- Na otwartej przestrzeni nie stój w bezpośrednim słońcu i pod słońce.
- Pamiętaj, że za dużo światła powoduje zbyt silny kontrast, za mało – utratę ostrości.



## PERSPEKTYWA

- W wersji klasycznej: trzymaj telefon 10–20 cm powyżej linii wzroku pod lekkim skosem.
- Zdjęcia en face rzadko się udają, zwykle korzystniej wychodzą zdjęcia z profilu lub półprofilu.
- Dobierz odpowiednio kąt, pamiętając o tym, że twarz nie jest symetryczna.
- Zwróć głowę w stronę światła, wzrok skup na obiektywie lub miejscu tuż pod nim.
- Śmiej się naturalnie – pamiętaj, że usta, oczy, brwi i czoło pokazują emocje.





## TŁO

- Zadbaj o nieprzypadkowe, ciekawe tło albo wybierz klasyczne płaskie tło, np. ścianę, drzwi, mural, żywopłot (wtedy nic nie odwraca uwagi od głównej postaci na zdjęciu).
- Zwróć uwagę na to, by zdjęcie nie wyglądało tak, jakby z głowy wystawały ci jakieś elementy tła. Czasem wystarczy przesunąć telefon o kilka milimetrów.
- Zastanów się, czy kolory tła pasują do twojego ubioru – zbyt intensywne tło może przekłamać kolorystykę zdjęcia, przez co twarz może wyjść sina albo bardzo różowa.



## EDYCJA

- Zrób kilka, kilkanaście zdjęć, wybierz najlepsze.
- Poddaj je edycji zgodnie ze wskazówkami z lekcji.
- Wstaw w wybrane przez siebie miejsce w mediach społecznościowych.




# 9. Projektowanie okładki książki i e-booka

- Etapy opracowywania projektu graficznego
- Budowa i funkcje okładki drukowanej oraz e-booka
- Przygotowanie projektu w edytorze grafiki

## ZADANIE NA START

**Poznasz książkę po okładce**

Wejdź na platformę **www.ted.com**, która gromadzi najciekawsze wykłady z cyklu TED Talks, i posłuchaj wykładu *Designing books is no laughing matter. OK, it is* (*Projektowanie książek to nic śmiesznego. No, może trochę*) wygłoszonego przez Chipa Kidda, słynnego amerykańskiego projektanta okładek. Z którymi tezami grafika się zgadzasz, a z którymi nie?



## OKŁADKI PUBLIKACJI TRADYCYJNYCH

Okładka i wkład, czyli zbiór kart ułożonych w konkretnej kolejności i połączonych na jednym boku (grzbiecie wkładu), to dwa elementy tworzące wielostronicową publikację poligraficzną. Okładka chroni wkład przed mechanicznymi uszkodzeniami, stanowi opakowanie publikacji, swoistą wizytówkę oraz źródło informacji dla potencjalnego czytelnika. Pod względem stylistyki powinna stanowić jednolitą całość wraz z layoutem i treścią wkładu. Na okładkę mogą być nakładane obwoluty i opaski.

**POSZUKAJ W INTERNECIE**

Sprawdź, czym jest ISBN, kto go nadaje i kiedy jest niezbędny.

Okładka składa się z czterech stron, z czego dwie mogą być niezadrukowane, oraz grzbietu, na którym zwykle umieszcza się tytuł, imię i nazwisko bądź samo nazwisko autora oraz logo wydawnictwa. Elementy na grzbiecie powinny zostać rozmieszczone w ten sposób, aby z boku książki leżącej na tylnej okładce można je było przeczytać prawidłowo, a nie do góry nogami. Szerokość grzbietu zależy od liczby kart tworzących wkład oraz od rodzaju i gramatury papieru.




**Rys. 1.** Pierwsza i czwarta strona okładki do oprawy miękkiej, z zaznaczonym grzbietem i liniami cięcia

Okładka może też być wzbogacona o tzw. skrzydełka. Na skrzydełku przednim zwykle umieszcza się streszczenie lub fragment książki, a na tylnym – fotografię i biogram autora.

Przed przystąpieniem do projektowania okładki książki warto zobaczyć, jak zrobili to inni, aby nie wyważać otwartych drzwi.




**Rys. 2.** Pierwsze strony okładki rozprawy *Książę* Niccola Machiavellego, znanej również pod tytułem *Traktat o Księciu*

### Ćwiczenie 1. Subiektywna ocena okładki

Poszukaj rankingu najlepszych bądź najciekawszych okładek książek wydanych w danym roku (taki ranking przygotowuje np. tygodnik „Polityka"). Wskaż, które trzy okładki najbardziej do ciebie przemawiają. Uzasadnij swój wybór.

## OKŁADKI PUBLIKACJI CYFROWYCH

W przypadku e-publikacji wystarczy zaprojektować pierwszą stronę okładki. Należy jednak zwrócić uwagę na wielkość grafiki, która nie może być zbyt duża. Trzeba też właściwie ustawić rozdzielczość. Zazwyczaj przyjmuje się, że do druku potrzeba przynajmniej 300 dpi, podczas gdy do publikacji elektronicznych – tylko 72 dpi. Coraz częściej jednak również okładki e-booków są projektowane w lepszej jakości.

### Ćwiczenie 2. Projekt okładki

Przygotuj w edytorze grafiki GIMP okładkę do książki i e-booka *Książę* Niccola Machiavellego, inną od prezentowanych powyżej. Tekst rozprawy znajdziesz w serwisie **https://wolnelektury.pl**.

1. Ustal format.
2. Zgromadź grafiki – własne lub gotowe. Pamiętaj o przepisach prawa autorskiego. Do portali zamieszczających grafikę na licencji Creative Commons CC należą m.in.
   - Wikimedia Commons – **https://commons.wikimedia.org**;
   - Openclipart Library – **https://openclipart.org**;
   - Pixabay – **https://pixabay.com**.
3. Rozmieść w edytorze elementy graficzne i tekstowe – zadbaj o czytelną kompozycję i odpowiednie liternictwo.

### Ćwiczenie 3. Prezentacja i ocena projektu

Porównaj swoją okładkę z okładkami kolegów. Wskaż kilka istotnych różnic i podobieństw. Co – twoim zdaniem – należy wziąć pod uwagę przy ocenianiu projektu?

## PODSUMOWANIE

- Okładka książki tradycyjnej i cyfrowej jest jej wizytówką.
- Pierwsza strona okładki zawiera tytuł i ewentualnie podtytuł, imię i nazwisko bądź samo nazwisko autora oraz zwykle nazwę wydawnictwa.
- Czwarta strona okładki zawiera ISBN, kod kreskowy, cenę książki oraz informacje o treści dzieła lub serii wydawniczej i często notę o autorze wraz z jego fotografią oraz fragmenty recenzji.

- Na grzbiecie okładki zwykle umieszcza się tytuł, imię i nazwisko bądź samo nazwisko autora oraz logo wydawnictwa. Szerokość grzbietu zależy od liczby kart tworzących wkład oraz od rodzaju i gramatury papieru.
- W przypadku e-booków projektuje się tylko pierwszą stronę okładki.

## PYTANIA SPRAWDZAJĄCE

1. Jakie są etapy przygotowania projektu graficznego?
2. Jakie elementy powinny zawierać okładki książki i e-booka?
3. Jakie kryteria należy wziąć pod uwagę przy ocenie projektu?

## PYTANIA QUIZOWE

1. Od czego zależy szerokość grzbietu okładki?
   **a.** Od numeru ISBN.
   **b.** Od roku wydania.
   **c.** Od liczby stron książki oraz rodzaju i gramatury papieru.
   **d.** Od długości tytułu.
2. Jaką rozdzielczość wystarczy ustawić do druku?
   **a.** 72 dpi.
   **b.** 300 dpi.
   **c.** Nie ma to znaczenia.
   **d.** 100 dpi.
3. Ile stron okładki projektuje się w przypadku e-booków?
   **a.** 4
   **b.** 3
   **c.** 2
   **d.** 1

## ZADANIA DODATKOWE

### Zadanie 1. Projekt okładki książki

Przygotuj ostatnią stronę okładki traktatu *Książę* Niccola Machiavellego w edytorze grafiki GIMP.

### Zadanie 2. Kreator okładek

Serwis **www.canva.com** to wszechstronny edytor online przeznaczony do projektowania logo, okładek, kartek urodzinowych, zaproszeń, ulotek, banerów, kolaży zdjęć, cytatów motywacyjnych, memów, plakatów, CV, a także wszelkiego rodzaju materiałów publikowanych w mediach społecznościowych. Przygotuj w nim okładkę ulubionego e-booka. Czy wygodniej pracuje ci się w tradycyjnym edytorze grafiki, czy w Canvie? Czym różnią się oba podejścia?



# EDYTOR TEKSTU

Umiejętność posługiwania się edytorem tekstu jest dziś przydatna zarówno w szkole, jak i na co dzień:

- podczas opracowywania referatów i sprawozdań lub pisania prac semestralnych czy dyplomowych,
- podczas pisania pracy licencjackiej czy magisterskiej,
- do przygotowania prostszych i bardziej skomplikowanych dokumentów, takich jak CV i list motywacyjny, ogłoszenie, ulotka czy e-book.

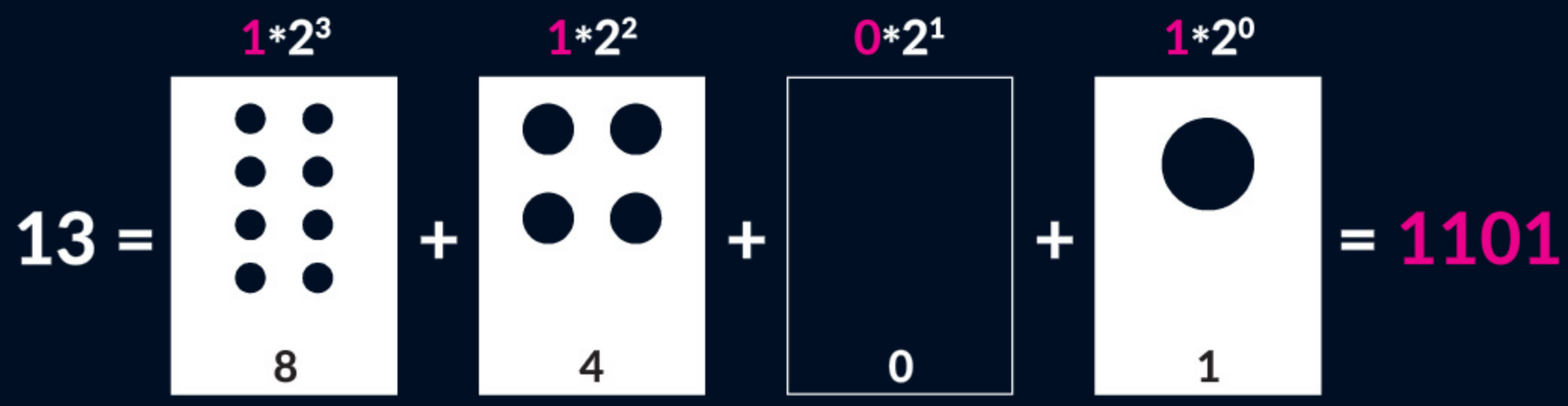
# SPOSÓB ZAPISU LICZB I TEKSTU

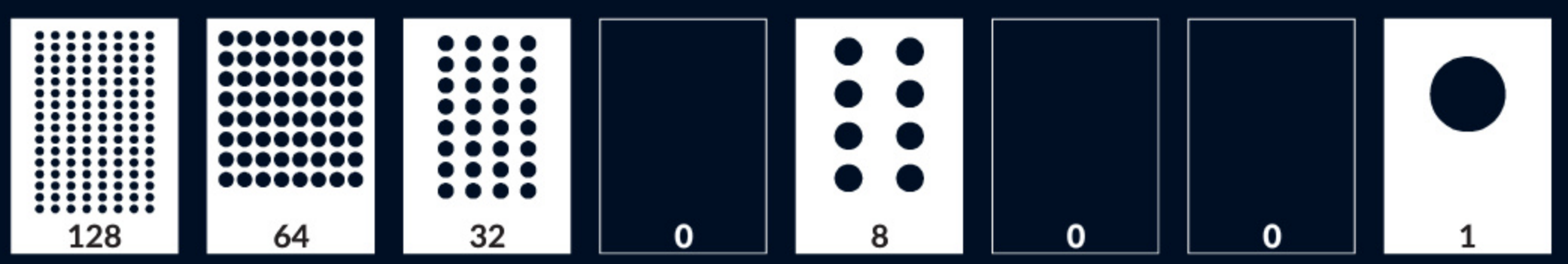
**Dane w komputerze są reprezentowane przez ciąg bitów w postaci cyfr 0 i 1. Binarny sposób zapisu informacji jest związany z tym, że komputer rozpoznaje tylko dwa stany napięcia: brak lub bardzo niskie (0) i wysokie (1). Dwa bity mogą przyjąć cztery stany: 00, 01, 10 i 11, trzy bity – osiem: 000, 001, 010, 011, 100, 101, 110, 111...**




## LICZBY

Kodowanie liczby naturalnej na *n*-bitach polega na zapisie liczby w systemie dwójkowym i na uzupełnieniu jej od lewej strony zerami do *n* pozycji.



## TEKSTY

Podstawą reprezentacji tekstów w komputerze jest zakodowanie wszystkich znaków w postaci liczb. System kodowania znaków **ASCII** przyporządkowuje liczbom z zakresu 0–127 litery alfabetu angielskiego, cyfry, znaki przestankowe oraz inne symbole i znaki sterujące. Każdy znak w kodzie ASCII jest przedstawiany jako siedmiobitowa liczba całkowita. Między wielką a małą literą jest tylko jeden bit różnicy:

A ↔ 1000001 ↔ 65

a ↔ 1100001 ↔ 97

Z ↔ 1011010 ↔ 90

z ↔ 1111010 ↔ 122




Kodowanie liter narodowych zapewniają standardy z rodziny Windows-1250, ISO 8859 oraz UTF-8. Znakom przyporządkowano liczby z zakresu 128–255. W standardzie ISO 8859-2:

Ą ↔ 10100001 ↔ 161

Ź ↔ 10101100 ↔ 172

### PRACA Z INFOGRAFIKĄ

1. Jaką największą liczbę można zakodować za pomocą 8 bitów?
2. W jakim zakresie kodów ASCII znajdują się małe litery alfabetu łacińskiego, a w jakim wielkie?

## WYBRANE ZNAKI W SYSTEMIE BINARNYM I DECYMALNYM

| ZNAK | BIN | DEC |
|---|---|---|
| spacja | 00100000 | 32 |
| ! | 00100001 | 33 |
| " | 00100010 | 34 |
| # | 00100011 | 35 |
| $ | 00100100 | 36 |
| % | 00100101 | 37 |
| & | 00100110 | 38 |
| ' | 00100111 | 39 |
| ( | 00101000 | 40 |
| ) | 00101001 | 41 |
| * | 00101010 | 42 |
| + | 00101011 | 43 |
| , | 00101100 | 44 |
| - | 00101101 | 45 |
| . | 00101110 | 46 |
| / | 00101111 | 47 |
| 0 | 00110000 | 48 |
| 1 | 00110001 | 49 |
| 2 | 00110010 | 50 |
| 3 | 00110011 | 51 |
| 4 | 00110100 | 52 |
| 5 | 00110101 | 53 |
| 6 | 00110110 | 54 |
| 7 | 00110111 | 55 |
| 8 | 00111000 | 56 |
| 9 | 00111001 | 57 |
| : | 00111010 | 58 |
| ; | 00111011 | 59 |
| < | 00111100 | 60 |
| = | 00111101 | 61 |
| > | 00111110 | 62 |
| ? | 00111111 | 63 |
| @ | 01000000 | 64 |
| A | 01000001 | 65 |
| B | 01000010 | 66 |
| C | 01000011 | 67 |
| D | 01000100 | 68 |
| E | 01000101 | 69 |
| F | 01000110 | 70 |
| G | 01000111 | 71 |
| H | 01001000 | 72 |
| I | 01001001 | 73 |

# 10. Podstawy edycji tekstu

- Formatowanie znaków, akapitów i strony
- Wykorzystanie tabulatorów
- Sprawdzanie poprawności pisowni

## ZADANIE NA START

**Korygowanie błędów**

Przeanalizuj poniższy tekst i wskaż usterki edycyjne oraz błędy innego typu.

> Spośród 10 znaków interpunkcyjnych kropkę , wielokropek , przecinek, średnik dwukropek, pytajnik i wykrzyknik stawia się bezpośrednio za wyrazem, po którym wystepują. Myślnik oddziela się od sąsiadujących wyrazów spacjami. W przypadku nawiasów i cudzysłowów ( znaków parzystych ) spację stawia się tylko przed znakiem otwierającym, a także po znaku zamykającym-o ile nie następuje po nim znak przestankowy. Cudzysłów stosuje do wydzielania cytatów orazoznaczania specyficznych użyć wyrazów i wyrażeń. W druku używa się najczęściej cudzysłowu apostrofowego „ ”.

## EDYTOR TEKSTU

Edytor tekstu, oprócz wprowadzania i edytowania tekstów, umożliwia ich formatowanie oraz wykonywanie innych, bardziej zaawansowanych czynności, takich jak:

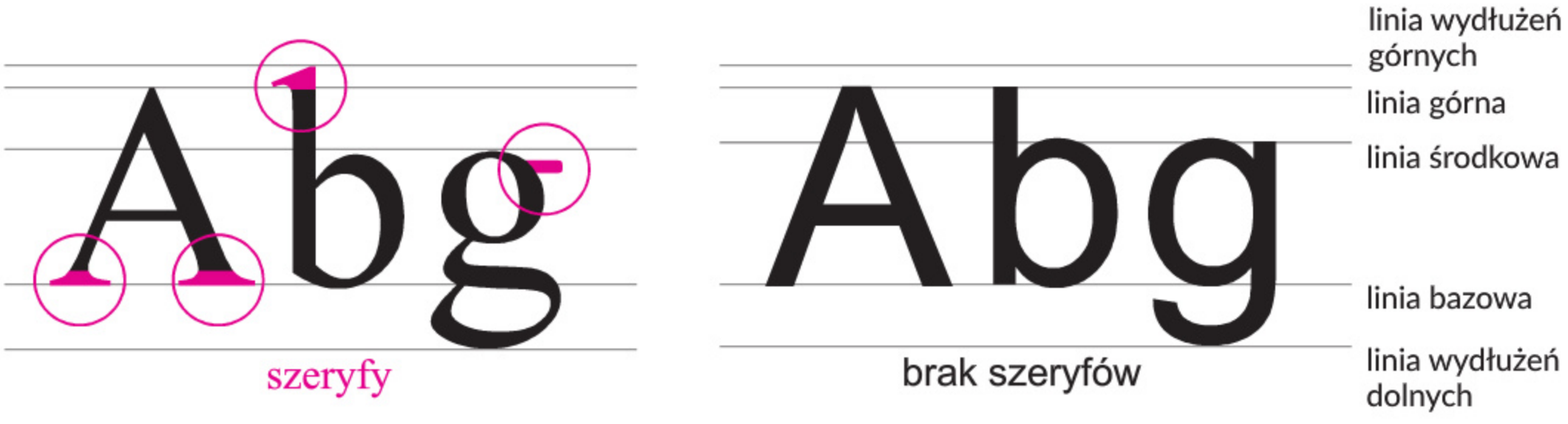
- osadzanie elementów graficznych statycznych i animowanych,
- wstawianie odnośników do stron internetowych,
- wstawianie tabel i wykresów przeznaczonych do edycji także w innych programach,
- chronienie hasłem,
- wyświetlanie historii zmian,
- praca w trybie rejestracji zmian,
- wysyłanie poczty elektronicznej,
- określanie języka,
- proponowanie wyrazów bliskoznacznych,

- sprawdzanie pisowni i gramatyki,
- przygotowywanie dokumentów w postaci PDF-ów lub stron WWW.

Do najpopularniejszych edytorów tekstów należą Word z pakietu Microsoft Office (dostępny w wersjach offline i online), Writer z pakietów Apache OpenOffice i LibreOffice oraz tworzone w chmurze Dokumenty Google, które umożliwiają również eksport pliku do formatu EPUB, coraz powszechniej stosowanego w czytnikach książek.

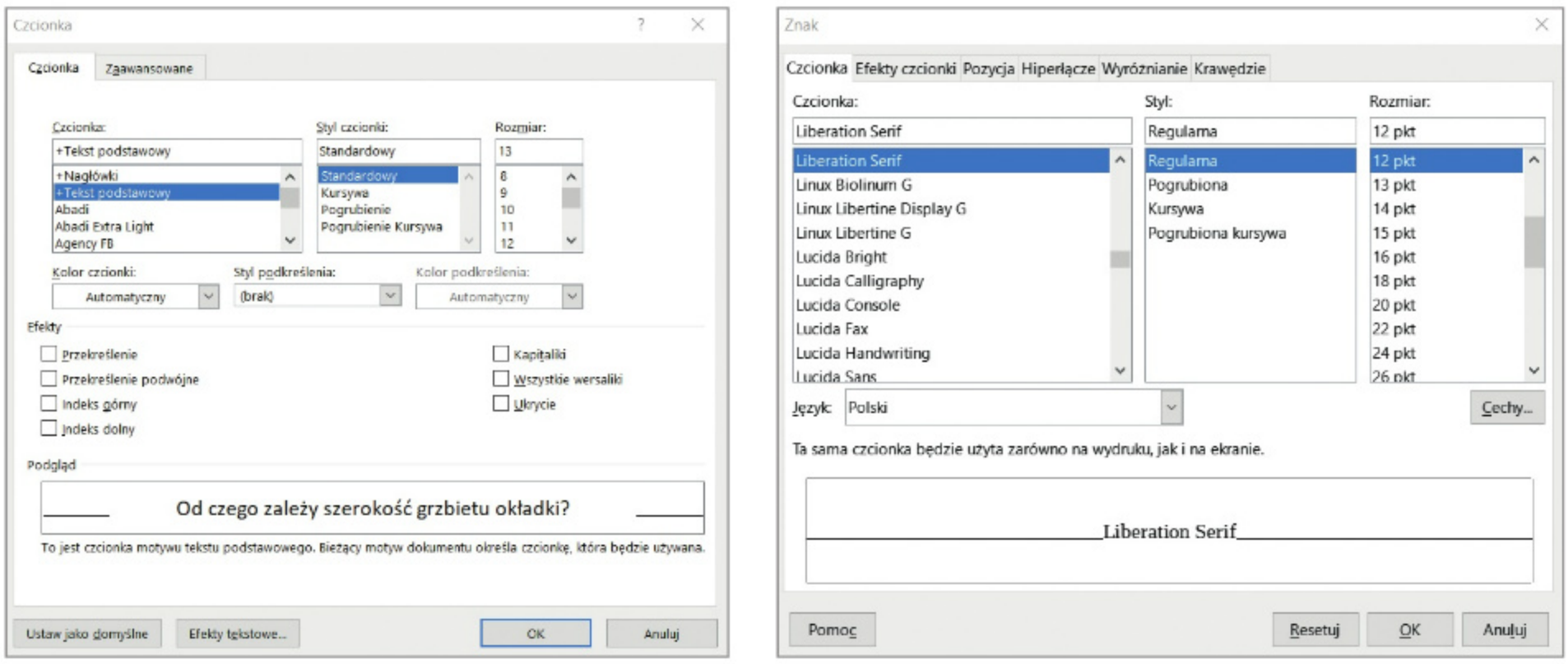
## CZCIONKA I AKAPIT

Czcionka to rodzaj nośnika pojedynczych znaków pisma drukarskiego. Obecnie tą nazwą określa się też font komputerowy. Czcionka zawiera obrazy wszystkich dostępnych znaków, zakodowane w postaci bitmapowej lub wektorowej. Może być szeryfowa (jak Times New Roman) lub bezszeryfowa (jak Arial).




**Rys. 1.** Pismo szeryfowe (Times New Roman) i bezszeryfowe (Arial) na liniach określających granice rysunku litery w wierszu

W zaawansowanych edytorach tekstu, takich jak Word czy Writer, można określić rodzaj czcionki, jej styl, rozmiar i kolor, dodać podkreślenia oraz różne efekty, np. przekreślenia, uwypuklenia czy kontury tekstu, zamienić zwykły tekst na WERSALIKI albo KAPITALIKI, a także ustawić odstępy między znakami. Zaznaczone opcje można zapisać jako domyślne.



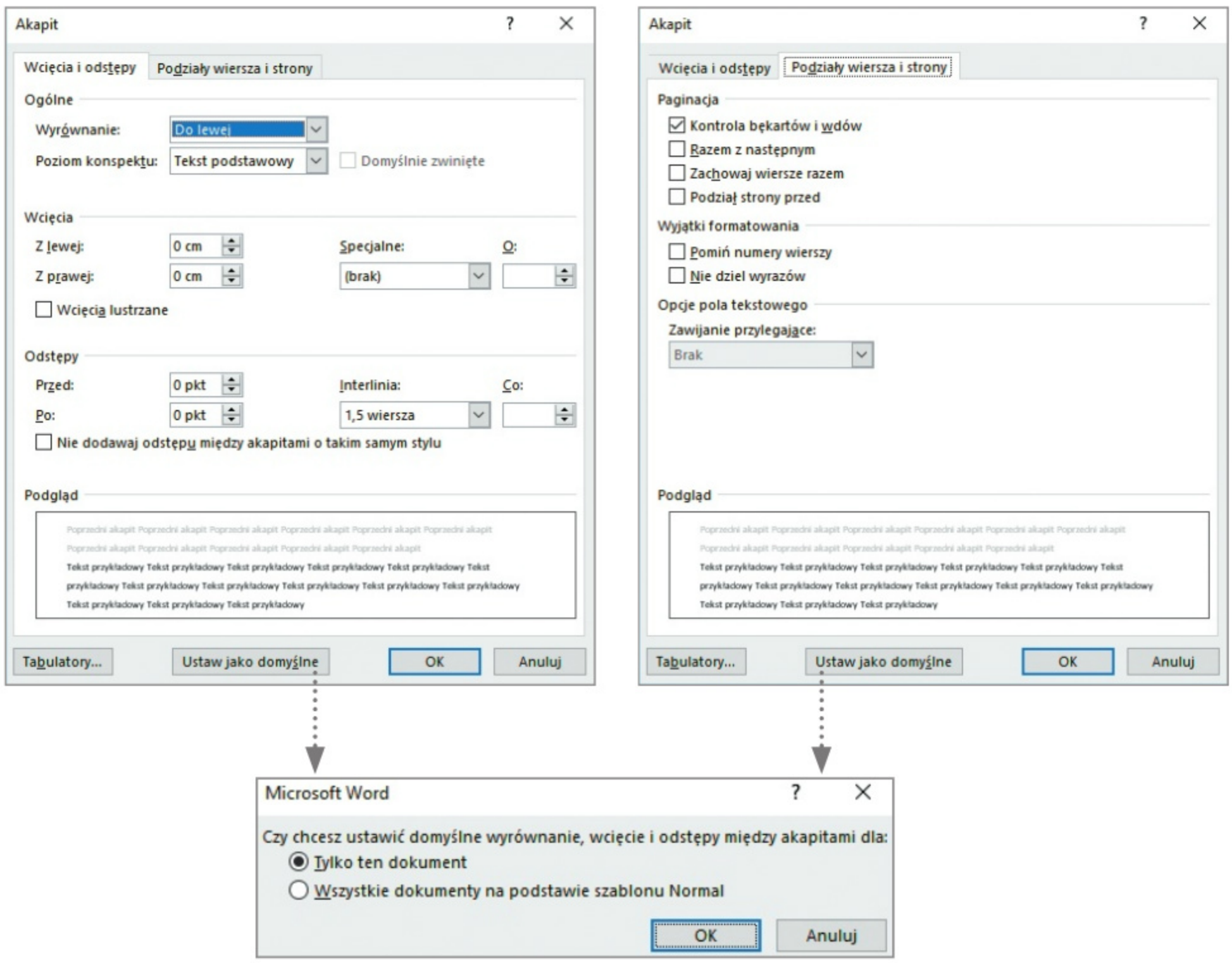

**Rys. 2.** Okna dialogowe narzędzi **Czcionka** (w edytorze Word) i **Znak** (w edytorze Writer)

**WIEM WIĘCEJ**

**Typografia** zajmuje się tworzeniem struktur i organizowaniem języka w wymiarze wizualnym – projektowaniem liter i innych znaków pisma drukowanego, ustalaniem relacji między poszczególnymi literami i znakami, a także zasadami graficznej kompozycji strony, która ma zostać wydrukowana.

Akapit to fragment tekstu stanowiący logiczną całość. Można go rozpocząć wcięciem pierwszego wiersza albo zaznaczyć zwiększonym odstępem między jednym a drugim fragmentem tekstu. Nie należy łączyć obu rodzajów wyróżnień akapitu.

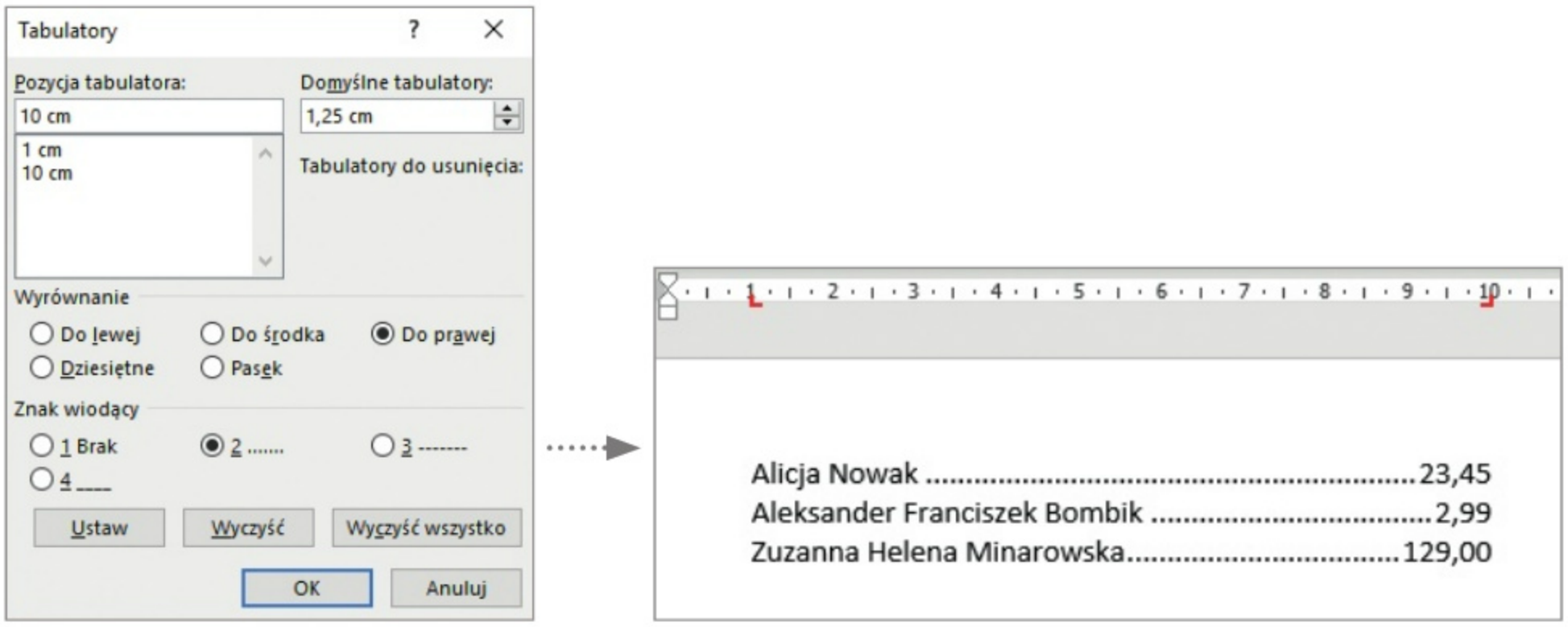
Częstym błędem popełnianym przy zmianie wyglądu tekstu jest formatowanie „ręczne", np. wyśrodkowywanie tekstu przez wstawianie spacji lub robienie odstępów między akapitami przez dodawanie pustych wierszy. Narzędzie **Akapit** pozwala ustawić m.in. wyrównanie zaznaczonego tekstu (do lewej, do prawej, do środka, wyjustowanie), wcięcia (z lewej, z prawej, pierwszy wiersz, wysunięcie), odstępy między wierszami tekstu, interlinię oraz tabulatory. Zaznaczone opcje można zapisać jako domyślne.




**Rys. 3.** Okno dialogowe **Akapit** w edytorze Word

Tabulatory wyświetlane na linijce poziomej wskazują sposób wyrównywania tekstu względem punktu tabulacji. W celu ustawienia tabulatora należy podać jego pozycję,

wybrać rodzaj wyrównania i znak wiodący. Zdefiniowane tabulatory można przesuwać w inne miejsce albo usuwać myszą z linijki, jeśli w danym akapicie nie są potrzebne.




**Rys. 4.** Ustawianie tabulatorów i wynik ich zastosowania w edytorze Word

Korzystanie z tabulatorów jest przydatne również przy tworzeniu zestawień.

| Nazwa | symbol | średnia punktów | liczba głosów |
|---|---|---|---|
| biegacz | BG1 | 2,51 | 144 |
| orbitrek | OR1 | 1,83 | 225 |
| wahadło | WH1 | 1,83 | 70 |
| narciarz | NC1 | 2,09 | 290 |
| wioślarz | WI2 | 1,71 | 171 |

**Rys. 5.** Wykorzystanie tabulatorów – wyrównanie do środka i paski (pionowe kreski)

## Ćwiczenie 1. Przygotowywanie szablonu dokumentu

Przygotuj szablon podania do Rady Rodziców XXX LO im. Smoka Wawelskiego w Krakowie (wzór poniżej). Sformatuj odpowiednio poszczególne części.

**.....** **Kraków, dnia .....**
**(imię i nazwisko)**

**.....**
**(klasa)**

**Rada Rodziców**
**XXX LO im. Smoka Wawelskiego**
**w Krakowie**

**PODANIE**

**Zwracam się z prośbą**

1. Ustal podstawowe elementy dokumentu oraz formatowanie czcionki i akapitu – zadbaj o czytelność i estetykę.
2. Podczas wpisywania kropek skorzystaj z tabulatorów ze znakami wiodącymi – po naciśnięciu klawisza **Tab** kursor przeniesie się we wskazane miejsce i wpisze kropki.

W Wordzie w oknie dialogowym **Akapit**, na karcie **Podział wiersza i strony**, w sekcji **Paginacja**, można ustawić automatyczną korektę wdów i bękartów, tj. eliminowanie jednowyrazowych wierszy pozostawionych na końcu akapitu oraz pojedynczych wierszy na początku strony lub kolumny. Samodzielnie należy zadbać o usuwanie pojedynczych liter na końcu wiersza (sieroty) oraz pojedynczych wierszy na końcu strony lub kolumny (szewc). Z kolei w edytorze Writer w oknie dialogowym **Akapit** na karcie **Przepływ tekstu** można ustawić zarówno kontrolę bękartów, jak i szewców.

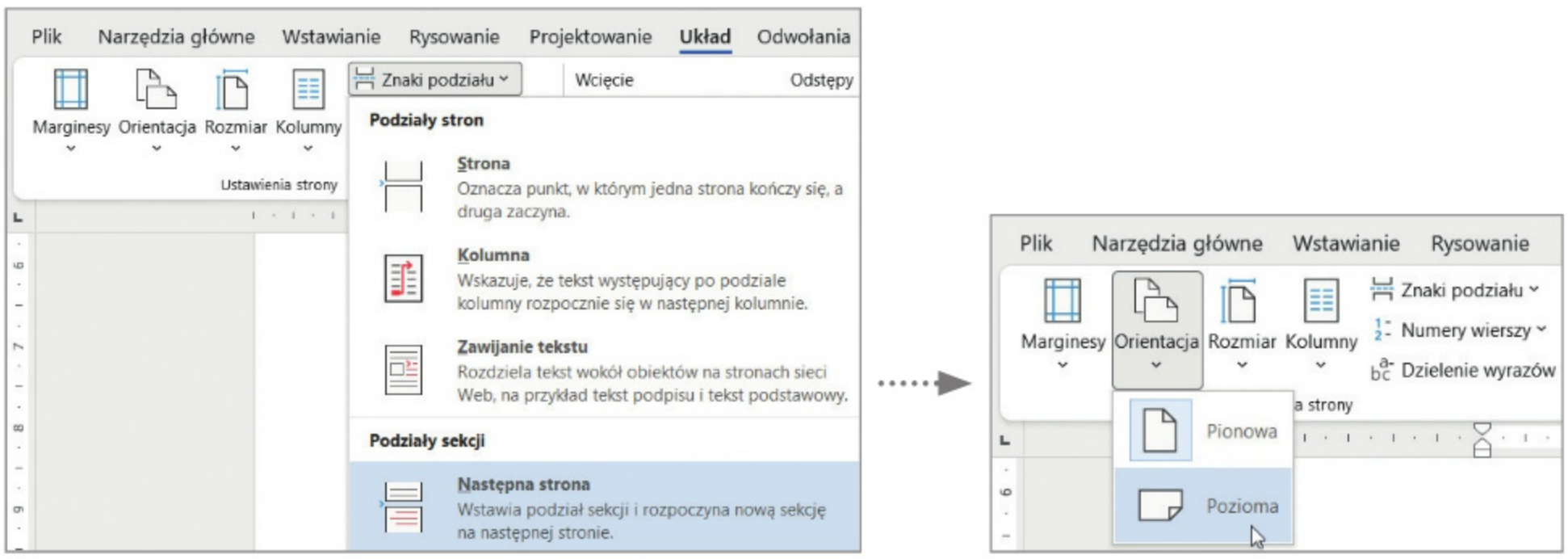



**Rys. 6.** Błędy, których należy się wystrzegać podczas edycji tekstu

## UKŁAD STRONY I OBRAMOWANIE

Zaawansowane edytory tekstu umożliwiają ustalenie orientacji stron w dokumencie, a także określenie rozmiaru strony i marginesów oraz ozdobienie strony lub tylko jej części – np. akapitu, tabeli, grafiki – obramowaniem.

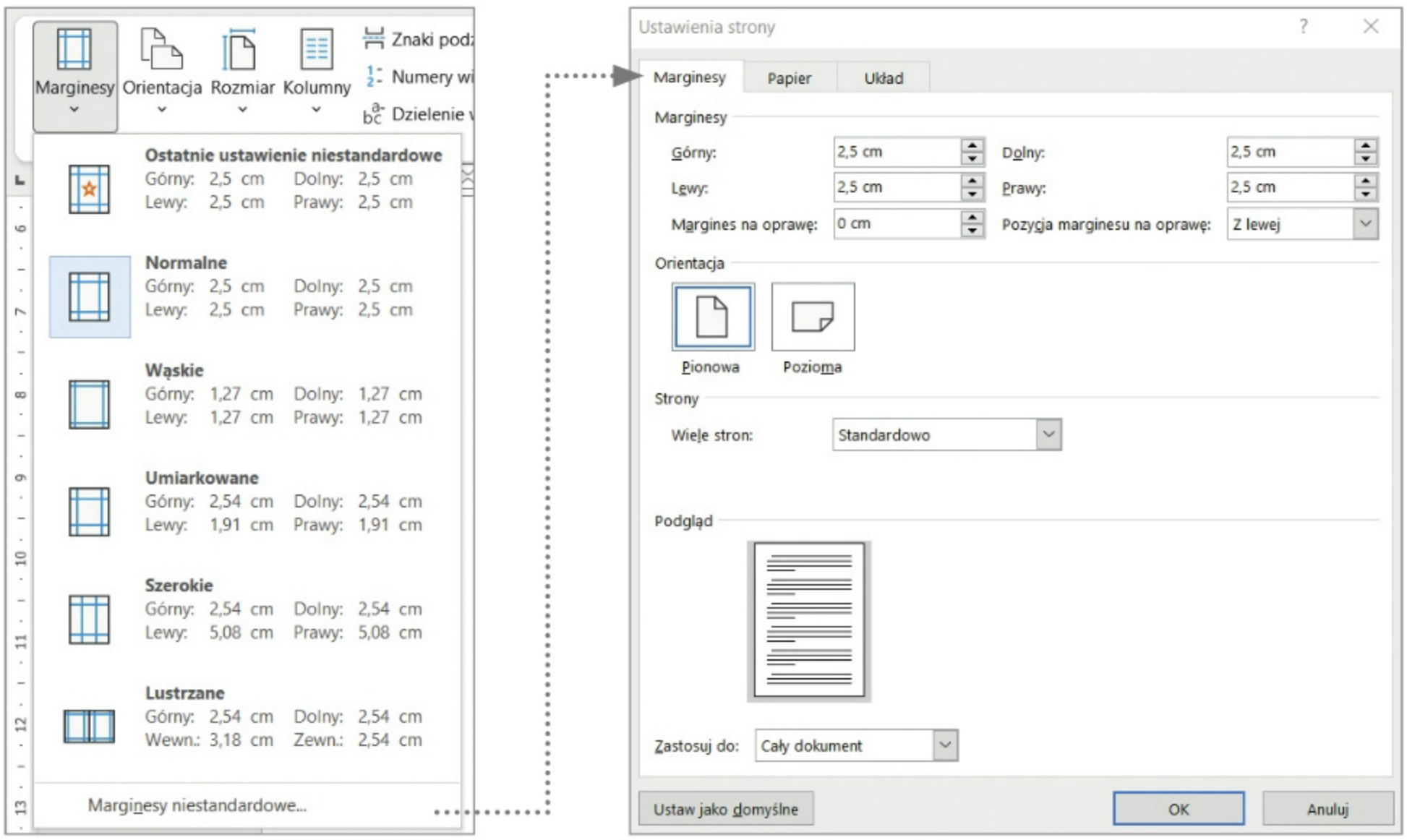
Domyślnie strony układane są w orientacji pionowej. Aby ustalić orientację poziomą dla całego dokumentu, w edytorze Word należy wybrać kartę **Układ** i w grupie **Ustawienia strony** rozwinąć listę **Orientacja**, a następnie wybrać opcję **Pozioma**. Aby wstawić wśród stron o orientacji domyślnej stronę w orientacji poziomej lub odwrotnie, w grupie **Ustawienia strony** należy najpierw rozwinąć listę **Znaki podziału** i wybrać opcję **Następna strona**, po czym rozwinąć listę **Orientacja** i wybrać odpowiednią opcję.




**Rys. 7.** Wstawianie strony w orientacji poziomej po stronie w orientacji pionowej w edytorze Word z pakietu Office 365

Standardowy format strony to A4 (210 mm × 297 mm). Jeśli zachodzi konieczność zastosowania innego formatu, np. zeszytowego A5, należy to zrobić przed rozpoczęciem tworzenia dokumentu. Wystarczy wybrać kartę **Układ**, a następnie w grupie **Ustawienia strony** rozwinąć listę **Rozmiar** i wskazać docelowy rozmiar papieru lub kliknąć opcję **Więcej rozmiarów papieru...** i ustalić rozmiar niestandardowy.

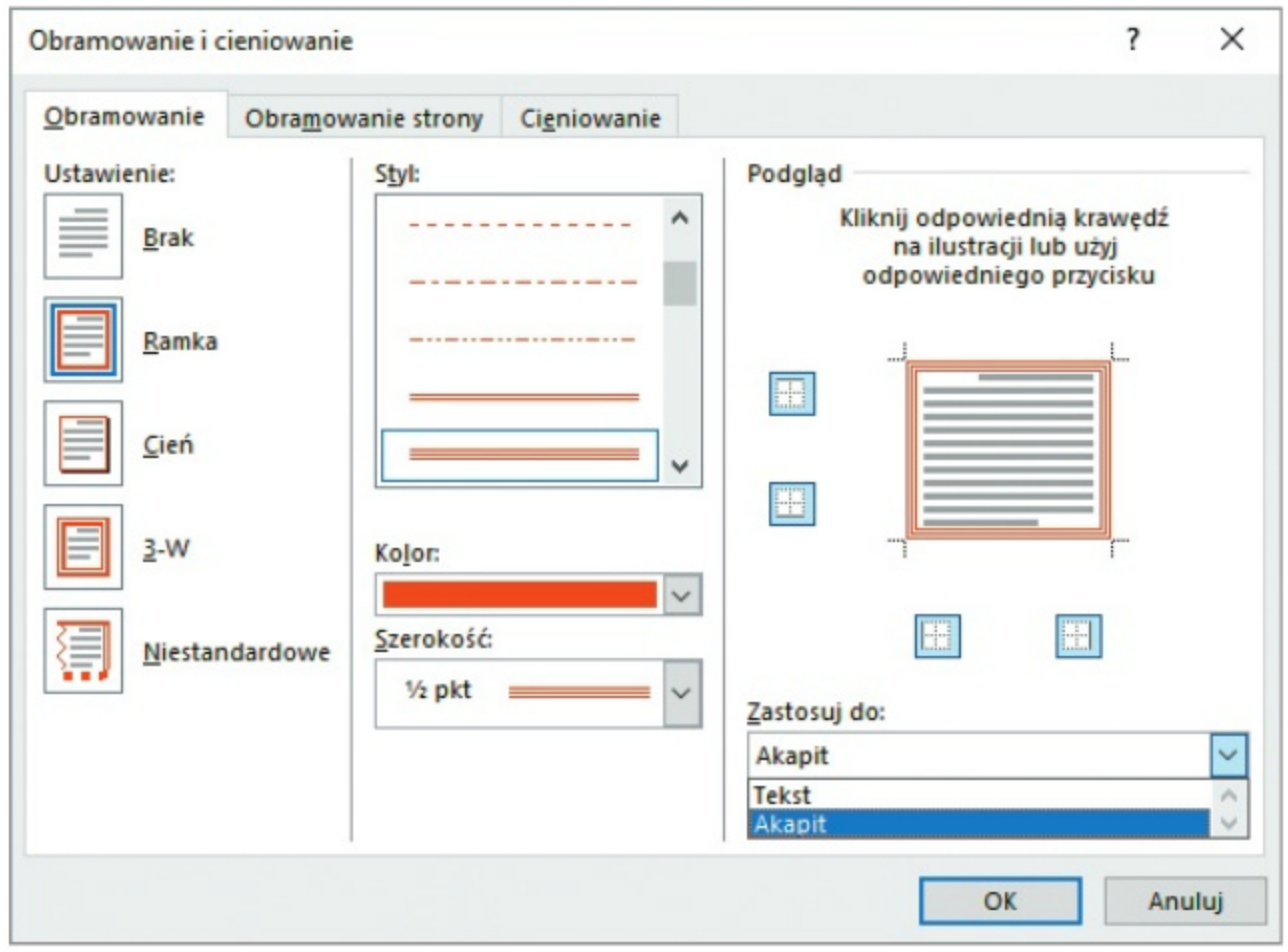
Domyślne marginesy (w Wordzie po 2,5 cm) również można zmienić w grupie **Ustawienia strony** z karty **Układ**. Po rozwinięciu listy **Marginesy** należy wskazać jedną z dostępnych opcji lub kliknąć opcję **Marginesy niestandardowe...** i wpisać odpowiednie wartości.




**Rys. 8.** Lista dostępnych marginesów i okno, w którym można ustawić marginesy niestandardowe w edytorze Word z pakietu Office 365

Aby powyższe modyfikacje wykonać w edytorze **Writer**, należy wybrać menu **Format**, narzędzie **Styl strony...** i kartę **Strona**.

Obramowanie zaznaczonego tekstu w edytorze Word umożliwia opcja **Obramowanie** na karcie **Narzędzia główne** w grupie **Akapit**. Aby obramować całą stronę, można wybrać opcję **Obramowanie → Obramowanie i cieniowanie… → Obramowanie strony** albo przejść na kartę **Projektowanie** i w grupie **Tło strony** kliknąć **Obramowania stron**. W edytorze Writer należy wybrać menu **Format**, a potem odpowiednio: albo narzędzie **Akapit...** i kartę **Krawędzie**, albo narzędzie **Styl strony...** i kartę **Krawędzie**.




**Rys. 9.** Obramowanie zewnętrzne zaznaczonego tekstu w Wordzie

## Ćwiczenie 2. Wprowadzanie tekstu

Napisz ogłoszenie o tym, że z serwisu WWW szkoły można pobrać szablon podania do Rady Rodziców (wzór poniżej).

**UWAGA!!!**

Koleżanki i Koledzy, informujemy, że szablon podania do Rady Rodziców można pobrać ze strony WWW szkoły (sekcja Dokumenty).

Samorząd Uczniowski
XXX LO im. Smoka Wawelskiego
w Krakowie

1. Zmień domyślną orientację strony na poziomą.
   **Układ → Ustawienia strony → Orientacja → Pozioma**

2. Wpisz tekst.

3. Obramuj stronę – wybierz jedną z dwóch ścieżek.
   **Narzędzia główne → Akapit → Obramowanie → Obramowanie i cieniowanie… → Obramowanie strony**

   lub

   **Projektowanie → Tło strony → Obramowania stron**

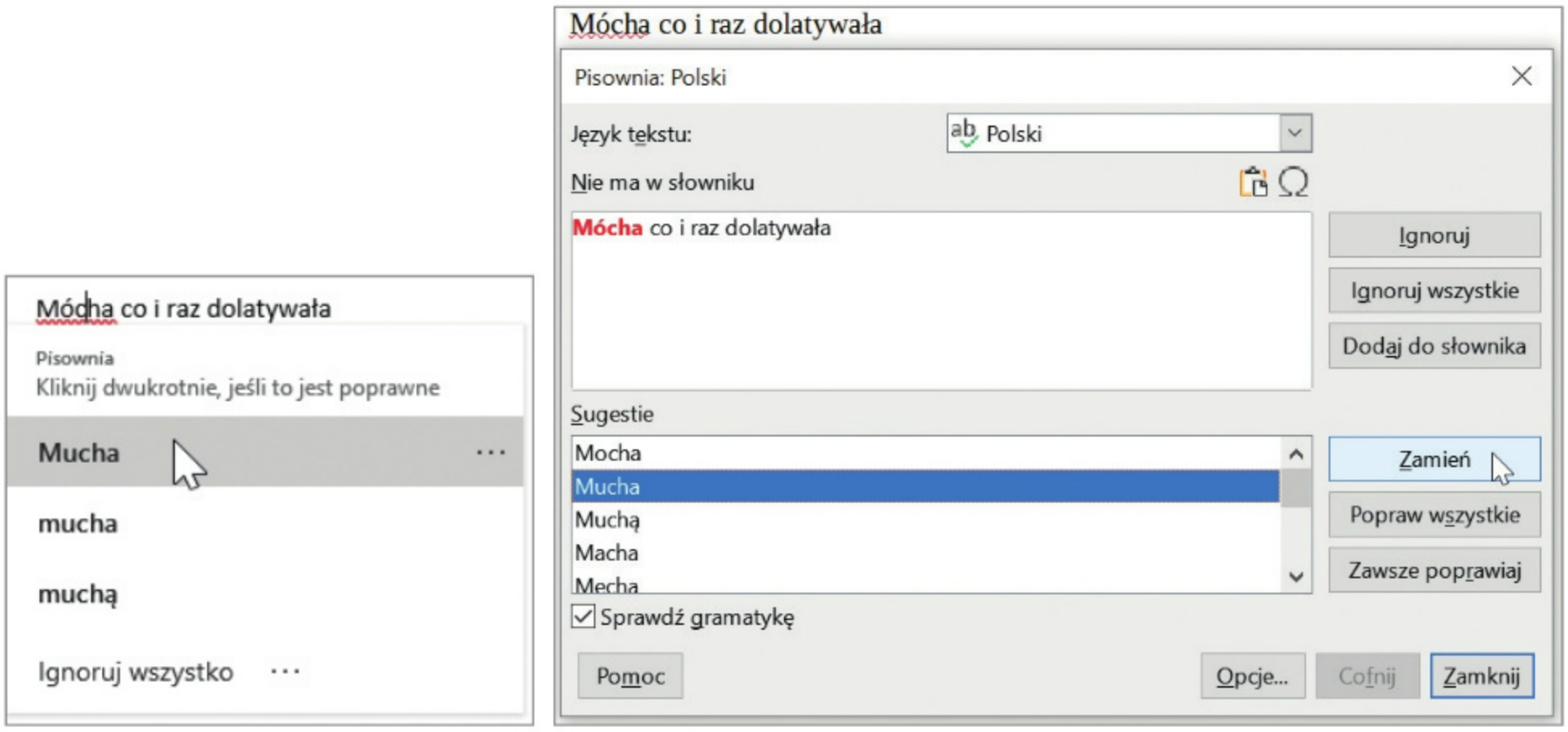
**WIEM WIĘCEJ**

Oto przydatne skróty klawiaturowe, służące do szybkiego i precyzyjnego zaznaczania tekstu.

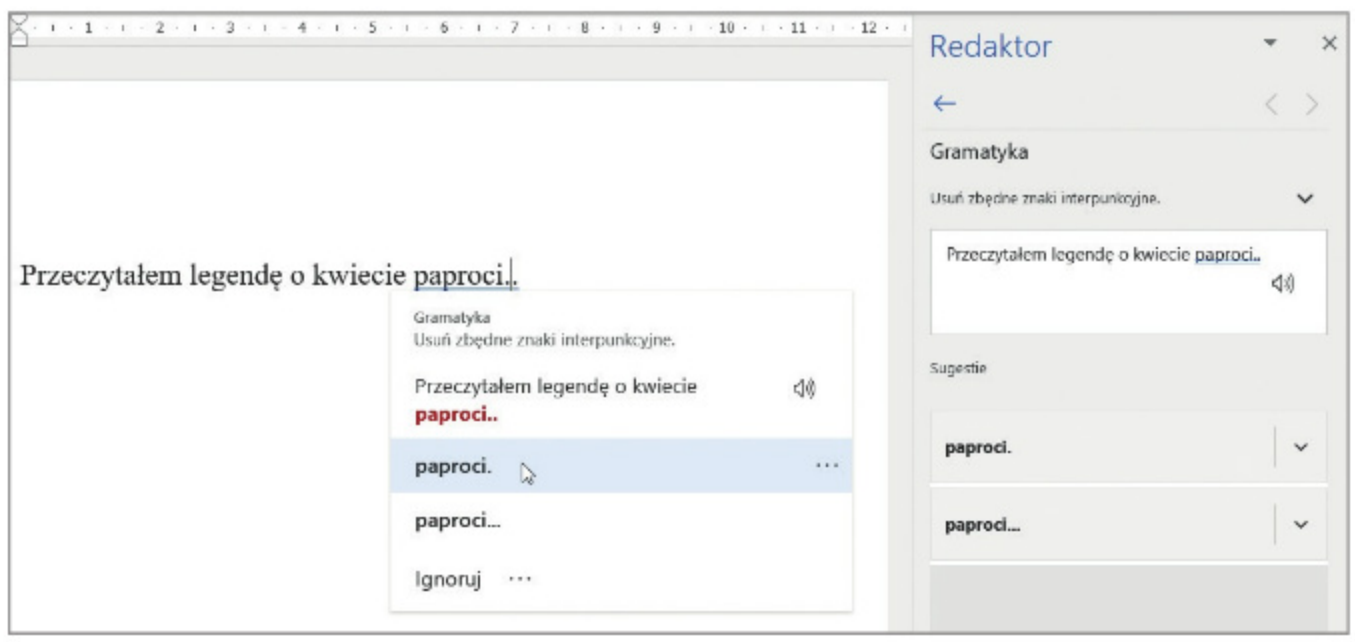
| | |
|---|---|
| **Ctrl+Shift+ → lub ←** | – zaznaczenie jednego słowa lub fragmentu tekstu (w zależności od położenia kursora) |
| **Ctrl+Shift+ ↑ lub ↓** | – zaznaczenie jednego akapitu |
| **Shift+Home** | – zaznaczenie tekstu do początku wiersza |
| **Ctrl+Shift+Home** | – zaznaczenie tekstu do początku dokumentu |
| **Shift+End** | – zaznaczenie tekstu do końca wiersza |
| **Ctrl+Shift+End** | – zaznaczenie tekstu do końca dokumentu |

## SPRAWDZANIE POPRAWNOŚCI PISOWNI

W programie Word na karcie **Recenzja** w grupie **Sprawdzanie** można znaleźć m.in. opcje umożliwiające sprawdzanie pisowni i gramatyki tekstu w dokumencie oraz sugerowanie synonimów. W edytorze Writer wszystkie opcje dotyczące pisowni umieszczono w menu **Narzędzia**.




**Rys. 10.** Błąd ortograficzny i możliwość poprawy w edytorach Word i Writer




**Rys. 11.** Błąd interpunkcyjny i możliwość poprawy w edytorze Word

Należy pamiętać, że nie zawsze zastrzeżenia co do poprawności pisowni lub gramatyki są uzasadnione, dlatego za każdym razem trzeba przemyśleć, co zrobić z „błędem", a nie bezkrytycznie akceptować poprawkę.




**Rys. 12.** Niesłuszne sugerowanie błędu

Korektę znaków interpunkcyjnych, np. cudzysłowów, można zautomatyzować. W przypadku edytora Word wystarczy wybrać **Plik → Więcej → Opcje → Sprawdzanie** i w **Opcjach Autokorekty** zaznaczyć pole umożliwiające zamianę podczas pisania cudzysłowów prostych ("") na drukarskie („"). Jeśli chodzi o myślniki, to edytor wyróżnia dywizy (łączniki), półpauzy i pauzy. Na klawiaturze umieszczono jedynie klawisz tworzący dywiz (-). Półpauzę (–) i sporadycznie potrzebną pauzę (—) można wstawić za pomocą opcji **Wstawianie → Symbol** lub kombinacji klawiszy **Ctrl**+**Num**-. Edytor wykonuje autokorektę, gdy napisze się kolejno: słowo, spację, dywiz, spację i słowo – wówczas dywiz zostaje zamieniony na półpauzę.

Aby wprowadzić tego typu modyfikacje w edytorze Writer, należy kliknąć menu **Narzędzia**, rozwinąć listę **Autokorekta**, wybrać polecenie **Opcje autokorekty...** i na kolejnych kartach skorygować domyślne ustawienia.

**WIEM WIĘCEJ**

**Słownik główny** edytora może nie obejmować niektórych wyrazów, np. nazw własnych, terminów technicznych, akronimów, specjalnych przypadków użycia wielkich i małych liter. Wyrazy niepoprawnie oflagowane przez moduł sprawdzania pisowni jako błędne warto dodać do domyślnego słownika niestandardowego.

- W programie Word w oknie sprawdzania pisowni i gramatyki należy kliknąć dla zaznaczonego wyrazu opcję **Dodaj** lub **Dodaj do słownika**.
- Podczas automatycznego sprawdzania pisowni i gramatyki należy kliknąć prawym przyciskiem myszy wyraz oznaczony czerwoną falistą linią i wybrać polecenie **Dodaj do słownika**.
- Można również dodać nowy słownik niestandardowy – zwykły plik tekstowy, w którym kolejne wyrazy zapisano w oddzielnych wierszach.

### Ćwiczenie 3. Weryfikacja poprawności tekstu

Sprawdź tekst ogłoszenia z ćwiczenia 2. Wykorzystaj w tym celu narzędzia wbudowane w edytor tekstu oraz wielojęzyczny moduł sprawdzania pisowni i stylu dostępny na stronie **https://languagetool.org/pl**. Porównaj oba poprawione teksty. Czy takie same błędy zostały skorygowane? Które narzędzie pozwala na przeprowadzanie bardziej zaawansowanych korekt?

## PODSUMOWANIE

- Edytor tekstu umożliwia wprowadzanie, edytowanie i formatowanie tekstu oraz wykonywanie innych czynności, m.in. zmianę orientacji strony, osadzanie elementów graficznych, wykresów i tabel, wstawianie hiperłączy, pracę w trybie rejestracji zmian, ozdabianie strony i znajdujących się na niej elementów za pomocą ramek, sprawdzanie pisowni i gramatyki.
- Tabulatory pozwalają na przygotowywanie zestawień, wyrównywanie tekstu oraz stosowanie znaków wiodących i pionowych kresek (tzw. pasków).
- Wbudowane w edytor słowniki umożliwiają sprawdzanie pisowni i gramatyki. Wyrazy niepoprawnie oflagowane przez moduł sprawdzania pisowni jako błędne można dodać do domyślnego słownika niestandardowego.

## PYTANIA SPRAWDZAJĄCE

1. W jaki sposób można wyrównać tekst po wstawieniu tabulatora?
2. W jaki sposób można obramować stronę, akapit oraz pojedynczy element graficzny?
3. Czy edytor zawsze poprawnie podpowiada korektę tekstu?

## PYTANIA QUIZOWE

1. Do czego służy format EPUB?
   **a.** Do zapisu książki możliwej do odczytania na wielu czytnikach.
   **b.** Do zapisu dokumentu przygotowanego do druku w profesjonalnej drukarni.
   **c.** Do przechowywania dużych spakowanych plików
   **d.** Do zapisu książki dostępnej za pośrednictwem stron WWW.
2. Do czego służy autokorekta?
   **a.** Do sprawdzania poprawności ortograficznej.
   **b.** Do sprawdzania poprawności stylistycznej.
   **c.** Do sprawdzania poprawności logicznej.
   **d.** Do automatycznego zamieniania jednego ciągu znaków na drugi.
3. Czym jest szewc?
   **a.** To pojedyncza litera na końcu wiersza.
   **b.** To samotny wiersz na końcu strony lub kolumny.
   **c.** To jednowyrazowy wiersz pozostający na końcu akapitu.
   **d.** To samotny końcowy wiersz na początku strony lub kolumny.

## ZADANIA DODATKOWE

### Zadanie 1. Ogłoszenie

Przygotuj ogłoszenie o sprzedaży garażowej (wzór poniżej). Do zestawienia oferowanych rzeczy oraz do obramowania tekstu wykorzystaj tabulatory.

# GARAŻÓWKA

**Lampa stojąca** ..... 123 zł

**Monitor LCD marki Star 15"** ..... 110 zł

**Resoraki (różne rodzaje, 123 sztuki)** ..... 2 zł/sztuka

### Zadanie 2. Plakat

Przygotuj zaproszenie do czynnego udziału w segregowaniu odpadów organizowanym na terenie szkoły. Dodaj elementy graficzne i tabelę z harmonogramem zbiórki.



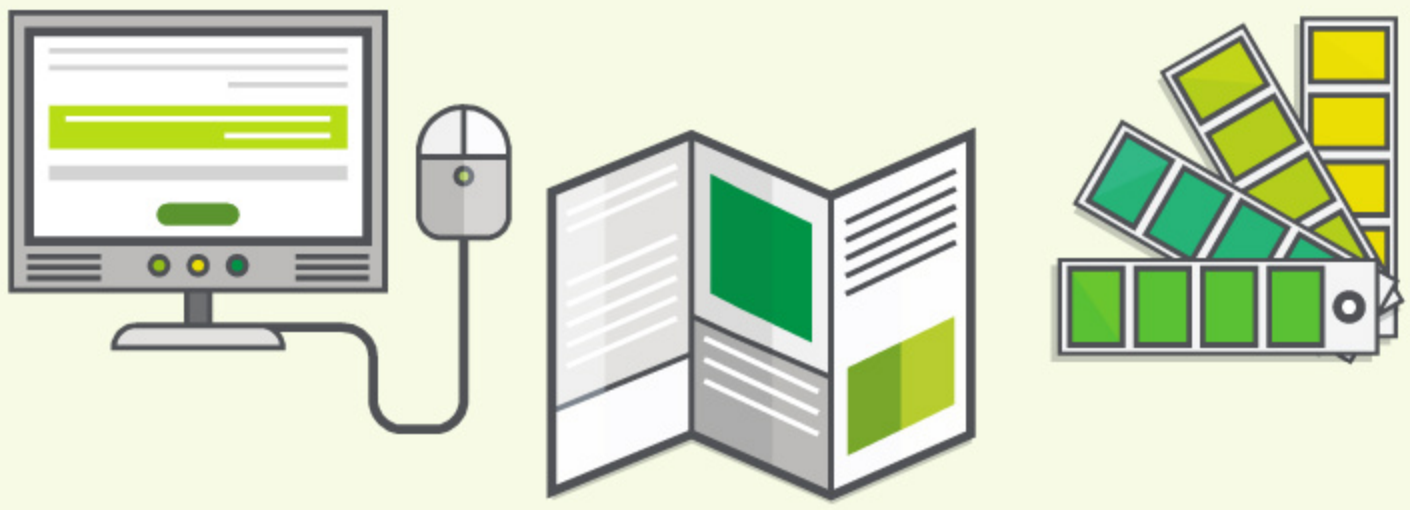
# 11. Przygotowanie publikacji do druku

- Podstawowe zasady łamania i składu tekstu
- Formatowanie tekstu za pomocą stylów
- Wstawianie rozbudowanych wzorów

## ZADANIE NA START

### Od projektu do druku

Podstawą stworzenia dobrego folderu, np. reklamowego, jest projekt. Liczy się pierwsze wrażenie. Odbiorcy najpierw patrzą na zdjęcia i ilustracje, następnie na główne hasła, a dopiero później skupiają się na właściwym tekście. Poszukaj informacji o tym, co oznacza skrót DTP i co się z tym wiąże.



## KOMPUTEROWA OBRÓBKA MATERIAŁU DO DRUKU

Przygotowanie publikacji do druku polega na:

- przygotowaniu szablonu (makiety) dokumentu;
- zredagowaniu tekstu i opracowaniu elementów graficznych;
- łamaniu – obróbce tekstu po redakcji w celu przygotowania wydruku;
- składzie – ostatecznym sformatowaniu tekstu, by uzyskał oczekiwany wygląd.

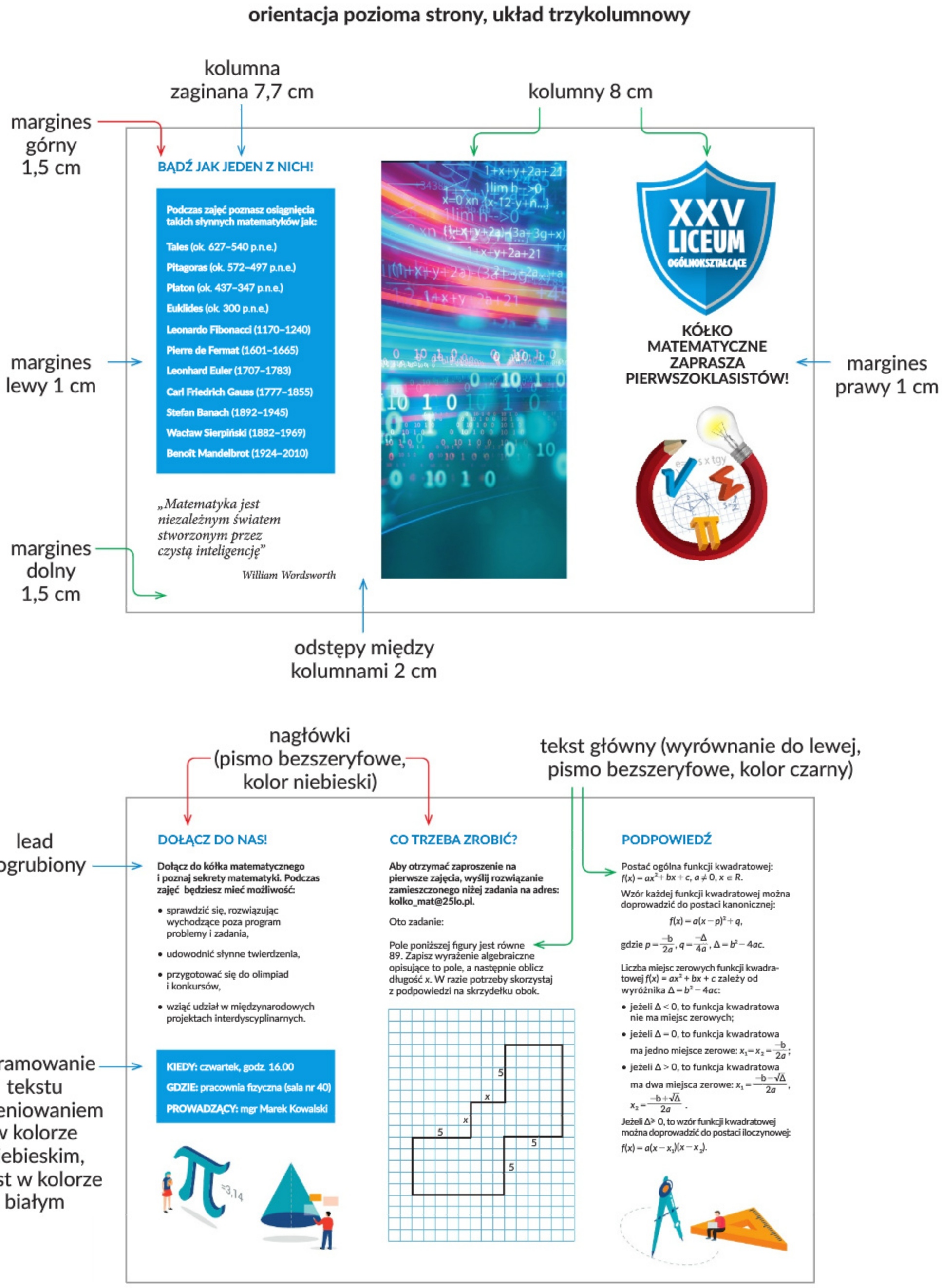
Proste publikacje można opracować w edytorze tekstu i zapisać w formacie PDF. Do zastosowań nieprofesjonalnych jest to zupełnie wystarczające. Przygotowanie tekstu do druku w specjalistycznym programie daje oczywiście większe możliwości edycyjne, ale jest trudniejsze i nierzadko wiąże się z dodatkowymi kosztami.

Publikacja powinna mieć określony layout – jednolity układ kompozycyjny strony – ponieważ

**POSZUKAJ W INTERNECIE**

Scribus jest bezpłatną aplikacją na wolnej licencji, służącą do przygotowywania broszurek, gazetek czy książek do druku. Poszukaj informacji na temat innych profesjonalnych programów do składania tekstów, także tych płatnych.

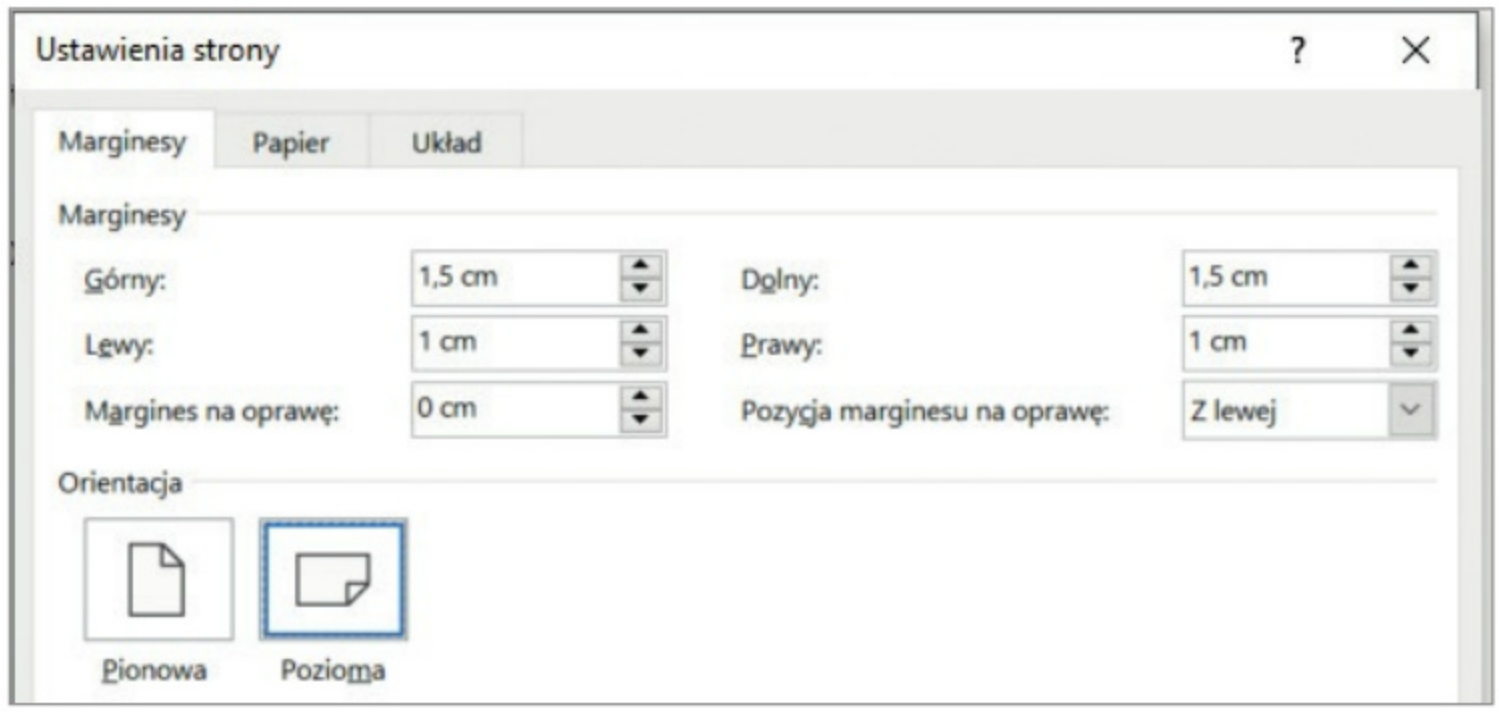
stałe elementy ułatwiają orientację. Podczas ustalania layoutu należy wziąć pod uwagę m.in. podział tekstu na kolumny i akapity, marginesy, krój i stopień pisma, interlinię, kolorystykę, powtarzające się elementy graficzne, paginację, grubość i kolor ramek.




**Rys. 1.** Przykładowy layout ulotki składanej na trzy w harmonijkę

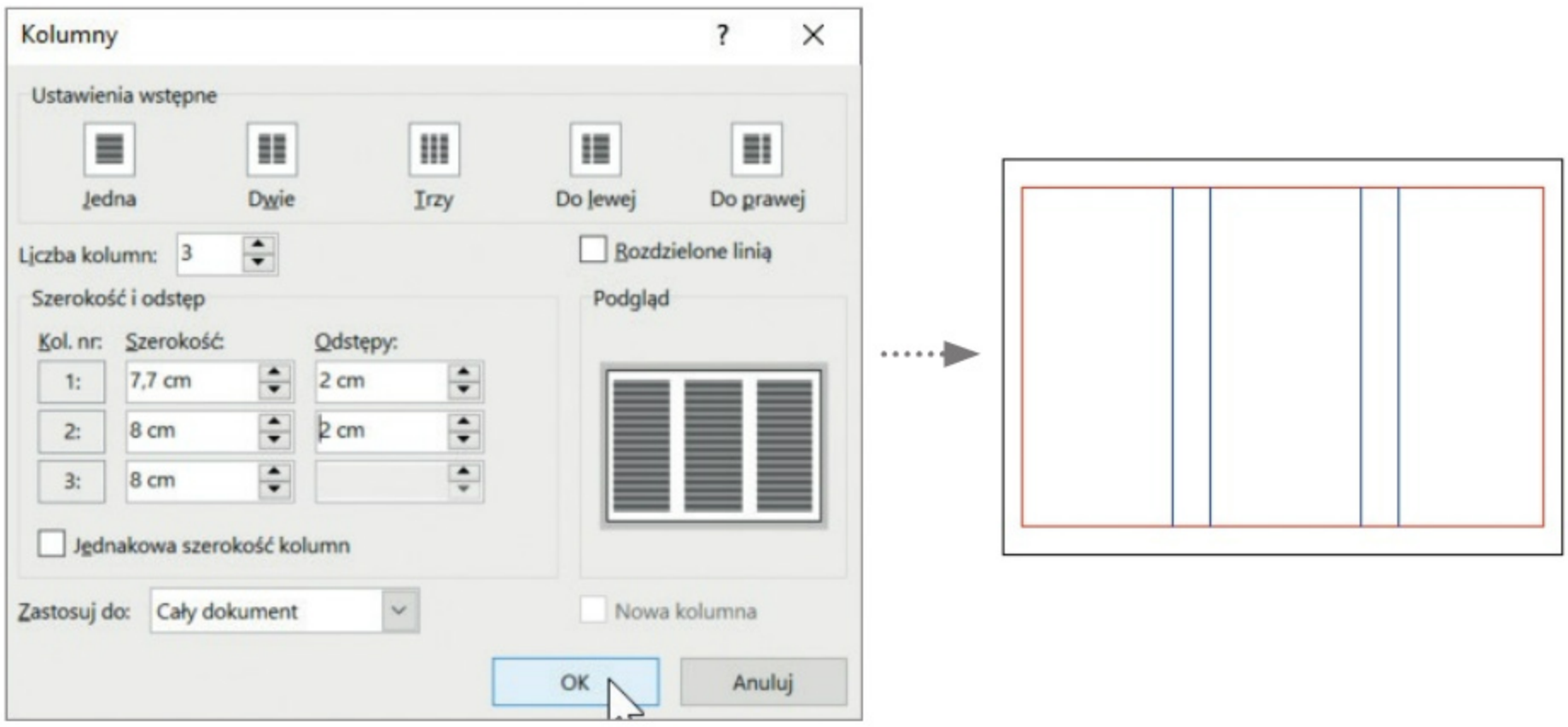
## TWORZENIE SZABLONU

Podczas ustalania wymiarów strony należy wziąć pod uwagę możliwości techniczne drukarki oraz funkcjonalność projektu. Na zwykłej drukarce najłatwiej drukuje się dokumenty w formacie A4, choć można oczywiście ustalić wymiary niestandardowe. W przypadku ulotki z rys. 1 należy zostawić domyślny rozmiar strony, a zmodyfikować ustawienia dotyczące orientacji, marginesów oraz kolumn.




**Rys. 2.** Ustawienia strony – marginesy i orientacja

Podczas ustalania szerokości kolumn należy uwzględnić sposób, w jaki dokument będzie składany. Jeśli strona nie jest w żaden sposób zginana, a kolumny pełnią tylko funkcję estetyczną, ich szerokość oraz odstępy między nimi można ustawiać dowolnie. W przypadku ulotki składanej na trzy w harmonijkę kolumna składana do środka powinna być nieznacznie węższa. Aby określić odpowiednią szerokość kolumn, na karcie **Układ** w grupie **Ustawienia strony** należy wybrać **Kolumny → Więcej kolumn**, odznaczyć domyślną opcję **Jednakowa szerokość kolumn** i uzupełnić wszystkie potrzebne parametry.




**Rys. 3.** Ustawienia strony dzielonej na trzy kolumny przy zadanych odstępach oraz założeniu, że lewe skrzydełko będzie składane do środka

### Ćwiczenie 1. Przygotowanie szablonu

Przygotuj w edytorze tekstu szablon dokumentu, który wykorzystasz do opracowania ulotki zachęcającej do uczestnictwa w kółku matematycznym. Zaproszenie ma być składane na trzy w harmonijkę.

1. Ustaw poziomą orientację strony.
2. Ustal marginesy, np. górny i dolny po 1,5 cm, lewy i prawy po 1 cm.
3. Ustal liczbę kolumn i ich szerokość – określ liczbę kolumn (3) oraz szerokość odstępu (np. 2 cm), a następnie odznacz opcję **Jednakowa szerokość kolumn** i odpowiednio zmniejsz szerokość pierwszej kolumny z lewej (np. o 3 mm). Szerokość pozostałych dwóch kolumn zostanie automatycznie zmodyfikowana.

**W PRAKTYCE**

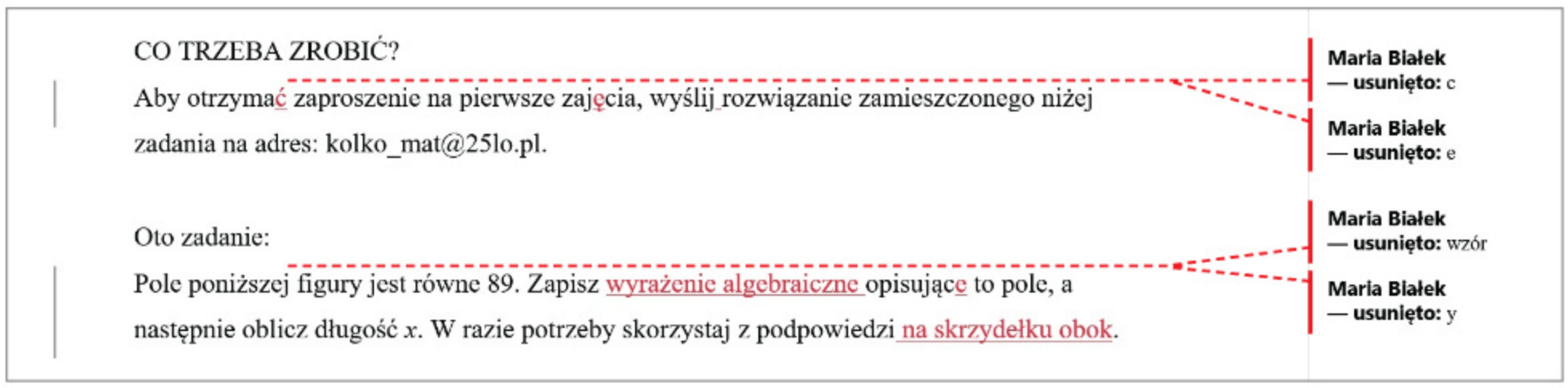
Użytkownicy smartfonów i tabletów z systemem Google Android i iOS mogą korzystać z edytora Microsoft Word w wersji mobilnej. Możliwe jest w nim nie tylko tworzenie, edytowanie oraz przeglądanie dokumentów, ale także stosowanie gotowych szablonów. Tekst można wpisywać za pomocą klawiatury albo dyktować.

## REDAGOWANIE I WSTAWIANIE TEKSTU

Tekst najlepiej przygotować w osobnym dokumencie, a następnie przekazać go innej osobie do sprawdzenia.

W pracy zespołowej lub redaktorskiej bardzo przydatnym narzędziem w edytorze tekstu jest tryb recenzji. Pozwala on na wprowadzanie komentarzy oraz widocznych zmian w tekście: usunięcia są oznaczane przekreśleniem, dodania – podkreśleniem, a zmiany wprowadzone przez poszczególnych autorów – różnymi kolorami.

Tego typu śledzenie zmian jest dostępne w różnych edytorach tekstu, również w tych działających w chmurze.



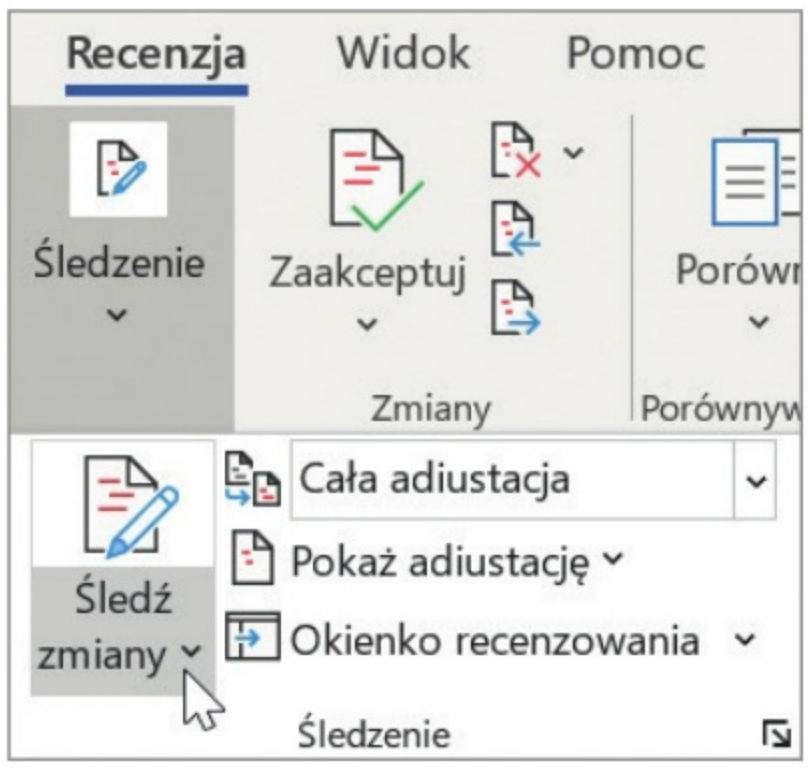

**Rys. 4.** Przykładowy dokument Worda z wprowadzonymi zmianami

Aby sugerować zmiany w dokumencie Google bez edytowania oryginalnego tekstu, właściciel pliku musi ustalić odpowiedni tryb pracy. Każdą z sugestii może zaakceptować lub odrzucić przez wybranie jednej z opcji ✓ ✕. Na koniec, gdy zatwierdzi propozycje zmian, zastąpią one oryginalny tekst.




**Rys. 5.** Sugerowane zmiany w dokumencie Google

Śledzenie zmian w programie Word jest dostępne na karcie **Recenzja** → **Śledzenie** → **Śledź zmiany**. Po włączeniu tej funkcji wszelkie usunięcia są oznaczane przekreśleniem, a dodania – podkreśleniem. Zmiany wprowadzone przez poszczególnych autorów są oznaczone różnymi kolorami. Po wyłączeniu śledzenia program Word przestaje zaznaczać zmiany, ale kolorowe podkreślenia i przekreślenia pozostają.

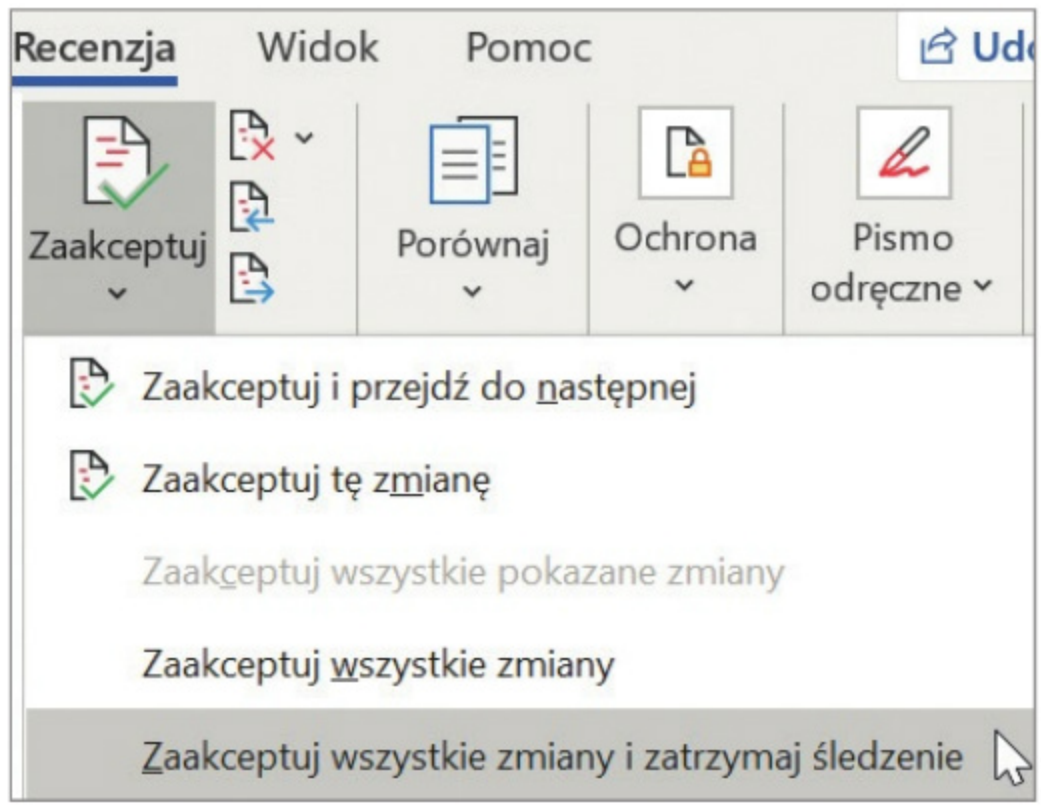



**Rys. 6.** Ustalenie trybu śledzenia zmian w edytorze Word

Jeśli kliknie się myszą pionową linię z lewej strony tekstu, można zamiennie ukryć lub pokazać zmiany w dokumencie. Kolor linii zmienia się na czerwony (zmiany ukryte) lub szary (zmiany pokazane). W trakcie przeglądania zmian w dokumencie można je akceptować pojedynczo lub wszystkie naraz.

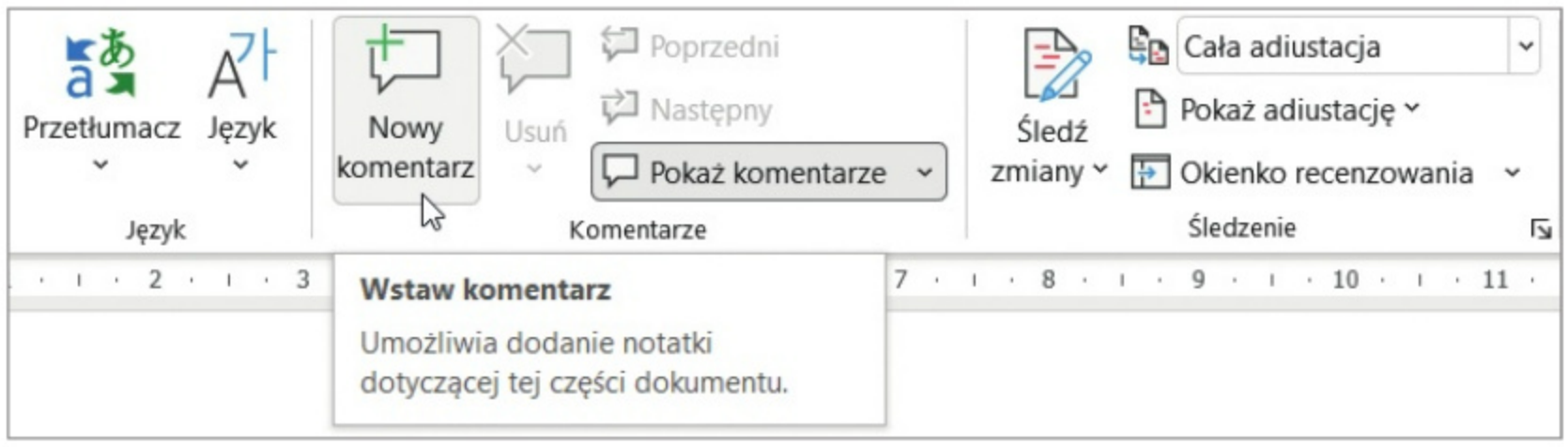
**W PRAKTYCE**

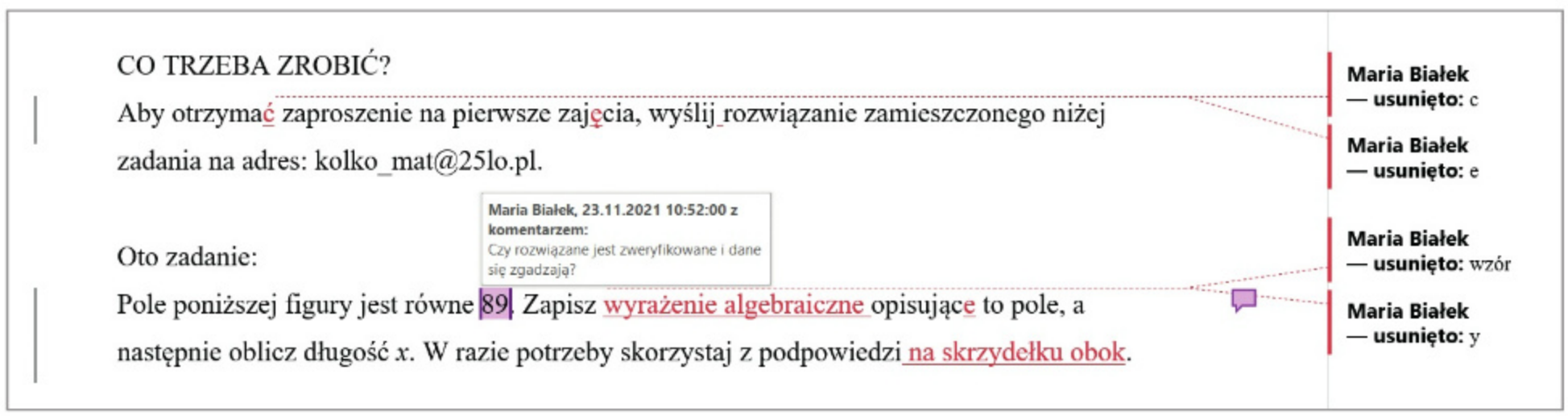
W edytorze Word można przetłumaczyć automatycznie akapit tekstu. Należy w tym celu zaznaczyć tekst i na karcie **Recenzja** w grupie **Język** wybrać polecenie **Przetłumacz**, a następnie sprawdzić, czy został wykryty właściwy język źródłowy, i wskazać język docelowy. Potem wystarczy nacisnąć **Wstaw**. Należy pamiętać, że jest to tłumaczenie maszynowe i może zawierać błędy.




**Rys. 7.** Akceptacja zmian w dokumencie

W tekście można dodawać także komentarze. Pozwalają one poznać punkt widzenia osoby pracującej nad dokumentem i umożliwiają wymianę opinii. Aby dodać komentarz, należy wskazać fragment tekstu, wybrać **Recenzja → Komentarze → Nowy komentarz**, a następnie wpisać treść komentarza i zatwierdzić operację. W komentarzu, oprócz treści, pojawią się nazwa lub imię autora i czas edycji.




**Rys. 8.** Wstawianie komentarza

Aby odpowiedzieć na komentarz, trzeba go kliknąć, a następnie wybrać opcję **Odpowiedz**. Potem wystarczy wpisać treść i ją zatwierdzić. Komentarze w tekście można pokazywać i ukrywać, drukować lub nie.

Nie trzeba znać wszystkich opcji programu na pamięć. Użytkownik aplikacji ma do dyspozycji pomoc kontekstową i ogólną. Należy jednak pamiętać, że jeżeli pracuje się lokalnie, to wówczas do dyspozycji jest więcej opcji niż w chmurze.

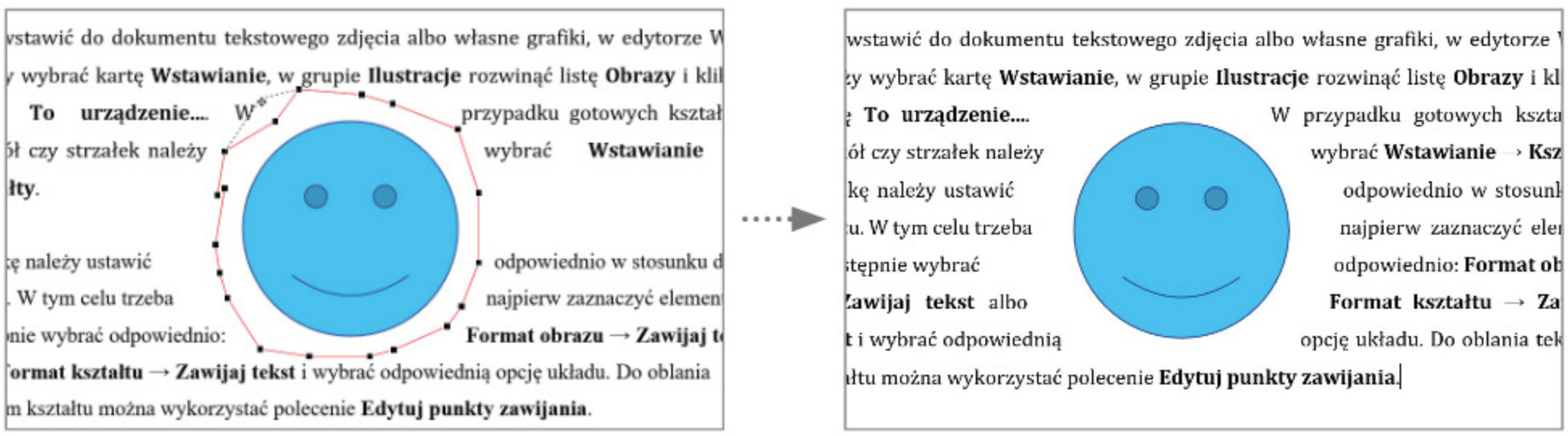
### Ćwiczenie 2. Uzupełnianie szablonu tekstem

Umieść przygotowany tekst w szablonie. Nie formatuj tekstu.

## WSTAWIANIE GRAFIKI

Aby wstawić do dokumentu tekstowego zdjęcia albo własne grafiki, w edytorze Word należy wybrać kartę **Wstawianie**, w grupie **Ilustracje** rozwinąć listę **Obrazy** i kliknąć opcję **To urządzenie...** Gotowe kształty, np. koła czy strzałki, można wstawić za pomocą opcji **Wstawianie → Kształty**.

Grafikę należy ustawić odpowiednio w stosunku do tekstu. W tym celu trzeba najpierw zaznaczyć element, a następnie wybrać odpowiednio: **Format obrazu → Zawijaj tekst** albo **Format kształtu → Zawijaj tekst**, po czym wybrać odpowiednią opcję układu. Do oblania tekstem kształtu można wykorzystać polecenie **Edytuj punkty zawijania**.



**Rys. 9.** Ustawianie grafiki względem tekstu z wykorzystaniem edycji punktów zawijania

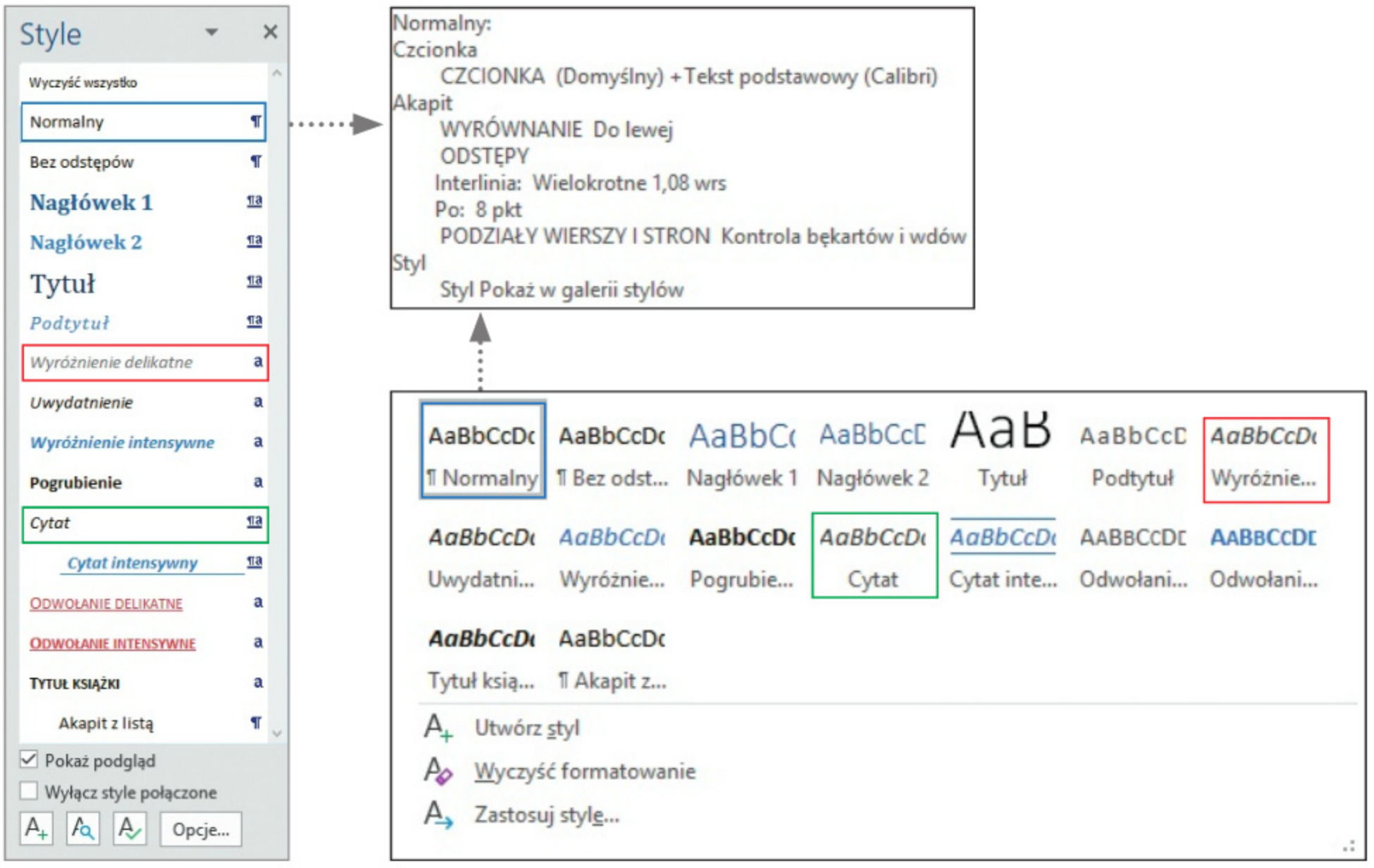
### Ćwiczenie 3. Uzupełnianie szablonu grafiką

Umieść przygotowane elementy graficzne w szablonie. W razie potrzeby ustal odpowiednio opcje układu.

## ŁAMANIE I SKŁAD

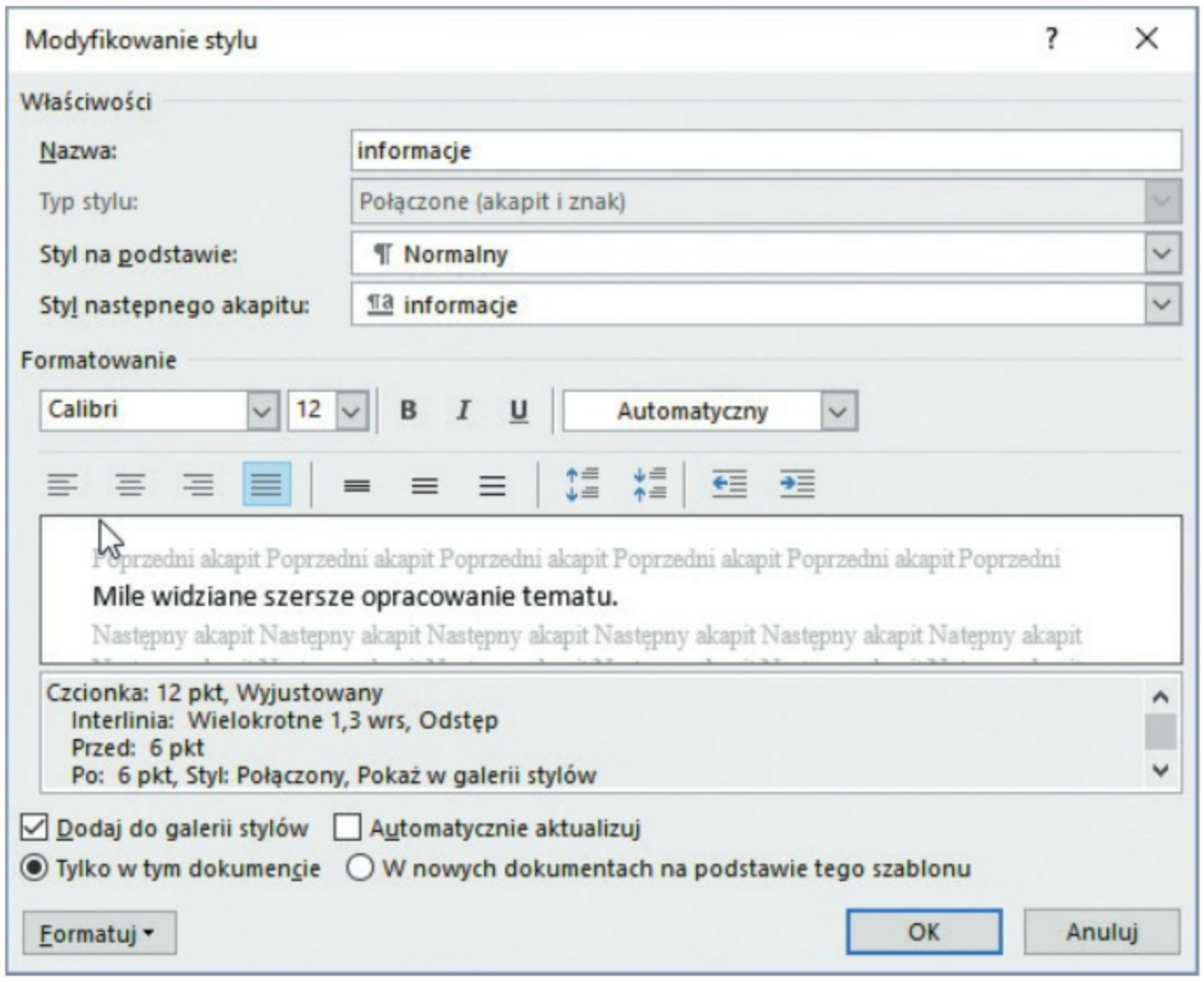
Po umieszczeniu tekstu oraz grafiki w szablonie i utworzeniu w ten sposób wstępnego obrazu publikacji należy przeprowadzić formatowanie.

W trakcie opracowywania założeń dotyczących formatowania czcionki oraz akapitu warto skorzystać z dostępnych stylów, które można dowolnie zmieniać, aby dostosować je do swoich potrzeb. Wybieranie stylu pozwala nadać tekstowi wiele atrybutów, m.in. określić rodzaj, styl i wielkość czcionki, zadbać o wyrównanie tekstu, interlinie i odstępy.




**Rys. 10.** Różne rodzaje i typy stylów

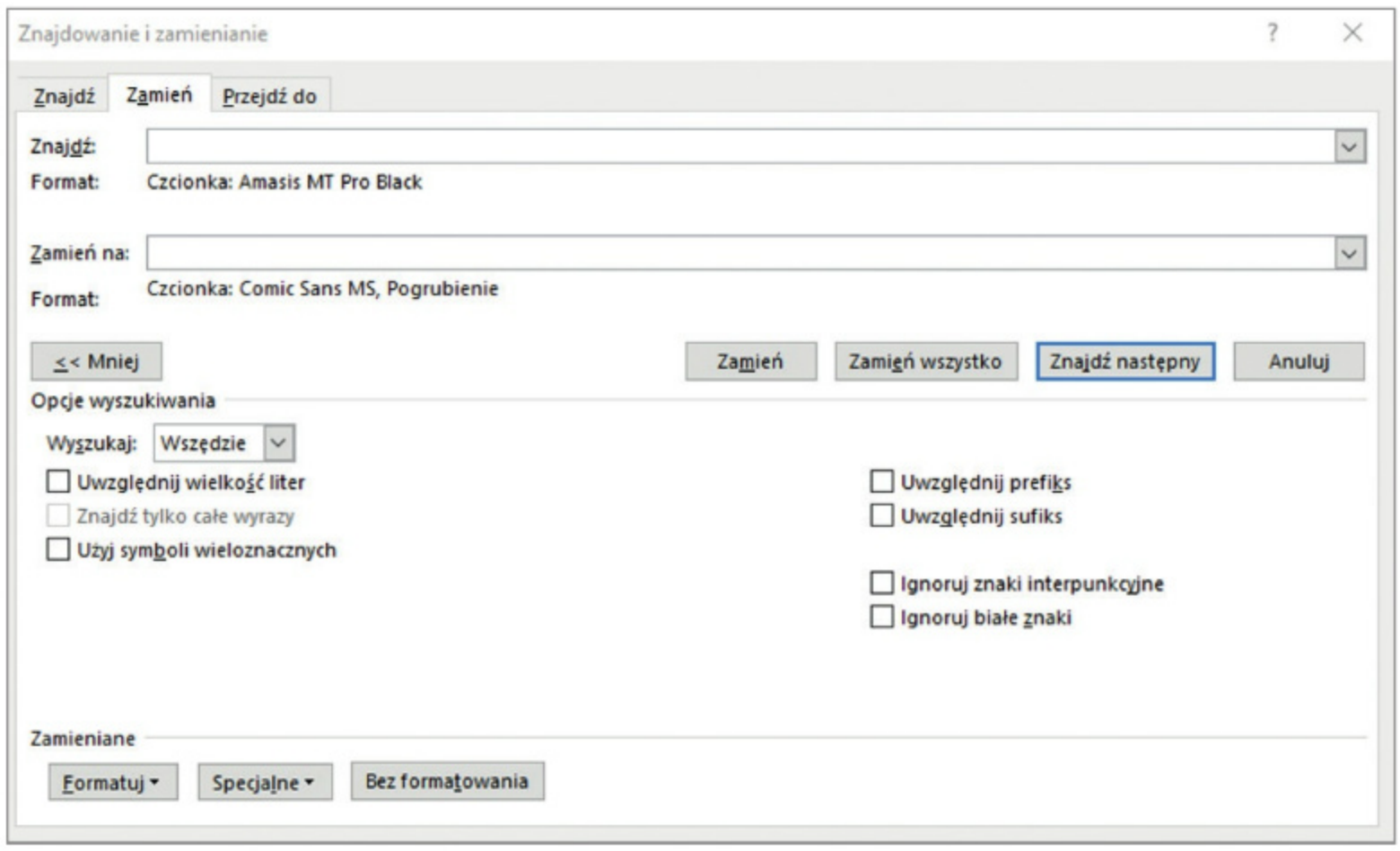
Modyfikacja stylu powoduje, że zmiana formatowania zachodzi automatycznie w całym tekście.




**Rys. 11.** Tworzenie nowego stylu na podstawie stylu Normalnego

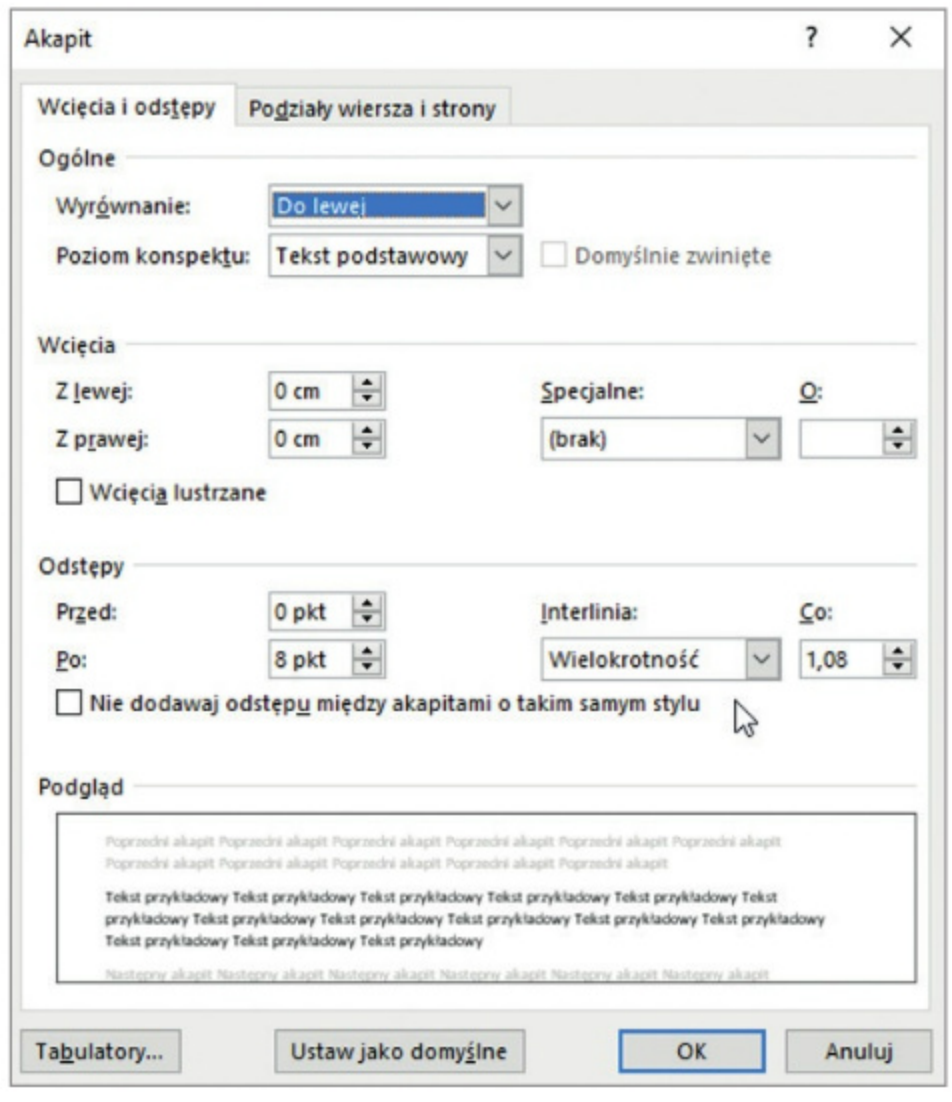
Jeśli konieczne jest zastąpienie wszystkich wystąpień danej czcionki inną, na karcie **Narzędzia główne** w grupie **Edytowanie** należy wybrać kolejno **Zamień** → **Więcej**,

a następnie umieścić kursor w oknie **Znajdź** i wybrać czcionkę, która ma być zastąpiona, przez kliknięcie **Formatuj → Czcionka**. Podobnie należy ustawić kursor w oknie **Zamień na:** i wskazać docelową czcionkę. Na koniec należy wybrać polecenie **Zamień wszystko**.




**Rys. 12.** Zamiana czcionki

Aby zmniejszyć lub zwiększyć odstęp między wierszami, na karcie **Narzędzia główne** w grupie **Akapit** należy wybrać ikonę **Interlinia** lub otworzyć okno dialogowe **Akapit** (mały kwadrat w prawym dolnym rogu), a następnie ustawić odpowiednią wartość w oknie **Interlinia**. Jeśli wiersze mają występować bardzo gęsto jeden po drugim, trzeba ustawić **Wielokrotność** i podać wartość mniejszą od 1, np. 0,8.




**Rys. 13.** Ustawianie interlinii

Jeżeli tekst jest wyjustowany, warto włączyć automatyczne dzielenie wyrazów, aby nie było w nim za dużo dziur. W tym celu na karcie **Układ** w grupie **Ustawienia strony** trzeba wybrać **Dzielenie wyrazów**, następnie zaznaczyć opcję **Automatyczne**. Należy pamiętać, żeby nie zostawiać na końcu wiersza wiszących liter. W edytorach istnieje możliwość wyświetlenia znaków niedrukowalnych. Ułatwia to analizę tekstu i naniesienie ewentualnych poprawek. Spacje są oznaczane kropkami, tabulatory – strzałkami, a koniec akapitu – znakiem ¶.

Aby automatycznie zamienić podwójne spacje na pojedyncze, nierozdzielające spacje na zwykłe oraz przejście do nowej linii na spację, należy skorzystać z narzędzia **Zamień** i wpisać w nim odpowiednie znaki do znalezienia i zastąpienia.

**ZAPAMIĘTAJ!**

**Twarda spacja** to znak, który wygląda jak zwykła spacja, ale nie pozwala w tym miejscu złamać tekstu. Wstawia się ją za pomocą kombinacji klawiszy **Ctrl+Shift+spacja**.
**Miękki Enter** to potoczne określenie przejścia do nowego wiersza w ramach tego samego akapitu. Ten znak wstawia się za pomocą skrótu klawiaturowego **Shift+Enter**.

### Ćwiczenie 4. Łamanie i składanie tekstu

Sformatuj dokument z ćwiczenia 3 i ustaw dzielenie wyrazów. Kolumnę z wzorami pozostaw pustą.

1. Dodaj wypunktowanie oraz obramowanie i cieniowanie tekstu.
2. Opracuj założenia dotyczące formatowania czcionki i akapitu – do tytułów w kolumnach oraz na stronie tytułowej wykorzystaj styl **Nagłówek 1**, a do tekstu w kolumnach zastosuj styl **informacje**, czyli zmodyfikowany styl **Normalny** (parametry – zob. rys. 11).
3. Ustaw dzielenie wyrazów.

## WSTAWIANIE WZORÓW

Aby wstawić wzory do tekstu, należy na karcie **Wstawianie** w grupie **Symbole** wybrać opcję **Równanie**. W miejscu, w którym stał kursor, pojawi się pole do wpisania równania, natomiast na wstążce wyświetli się karta **Równanie**.

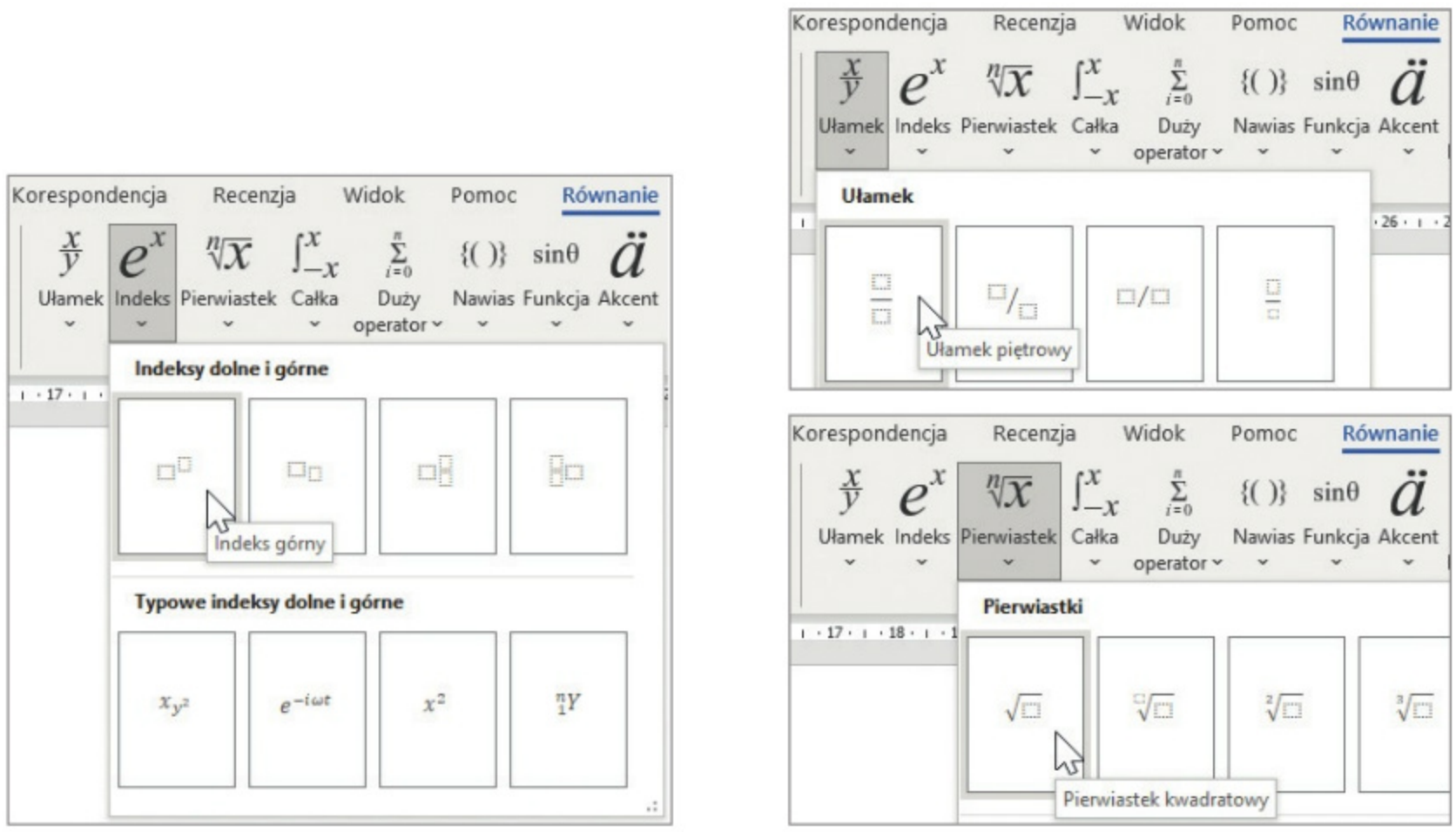



**Rys. 14.** Pole do wpisywania równania

Aby wpisać potęgę, należy na karcie **Równanie** w grupie **Struktury** wybrać formant **Indeks** – rozwinie się lista umożliwiająca zbudowanie odpowiedniej struktury potęgi. Następnie trzeba kliknąć **Indeks górny** i po wybraniu odpowiedniego zapisu zaznaczyć

myszą wybrane pole do uzupełnienia. Aby przenieść się z jednego pola do drugiego, wystarczy nacisnąć →. Ponowne naciśnięcie strzałki zakończy edycję potęgi.

Do wpisania ułamka zwykłego należy wybrać formant **Ułamek** oraz wskazać odpowiedni zapis ułamka. Wstawianie pierwiastków oraz wykonywanie pozostałych operacji i struktur przebiega podobnie jak w przypadku indeksów.




**Rys. 15.** Wstawianie indeksów, ułamków i pierwiastków do równania

Podczas wpisywania tekstu można korzystać z klawiatury lub z tablicy podstawowych operacji matematycznych, aby wstawić np. znaki działań.

## Ćwiczenie 5. Wstawianie wzorów

W ostatniej kolumnie ulotki z ćwiczenia 4 wpisz poniższe informacje na temat funkcji kwadratowej. Wykorzystaj edytor równań.

**Funkcja kwadratowa**
Postać ogólna funkcji kwadratowej: $f(x) = ax^2 + bx + c$, $a \neq 0$, $x \in R$.
Wzór każdej funkcji kwadratowej można doprowadzić do postaci kanonicznej:

$$f(x) = a(x-p)^2 + q, \text{ gdzie } p = \frac{-b}{2a}, q = \frac{-\Delta}{4a}, \Delta = b^2 - 4ac.$$

Liczba miejsc zerowych funkcji kwadratowej $f(x) = ax^2 + bx + c$ zależy od wyróżnika $\Delta = b^2 - 4ac$:

- jeżeli $\Delta < 0$, to funkcja kwadratowa nie ma miejsc zerowych;
- jeżeli $\Delta = 0$, to funkcja kwadratowa ma jedno miejsce zerowe: $x_1 = x_2 = \frac{-b}{2a}$;
- jeżeli $\Delta > 0$, to funkcja kwadratowa ma dwa miejsca zerowe: $x_1 = \frac{-b-\sqrt{\Delta}}{2a}$, $x_2 = \frac{-b+\sqrt{\Delta}}{2a}$.

Jeżeli $\Delta \geqslant 0$, to wzór funkcji kwadratowej można doprowadzić do postaci iloczynowej:

$$f(x) = a(x - x_1)(x - x_2).$$

## PODSUMOWANIE

- Łamanie i skład prostych publikacji można wykonać w edytorze tekstu typu Word czy Writer.
- Przez nadanie tekstowi stylu określa się wiele atrybutów, m.in. rodzaj, styl i stopień czcionki, wyrównanie tekstu, interlinie i odstępy. Styl można dowolnie zmieniać, aby dostosować go do swoich potrzeb. Modyfikacja stylu powoduje, że zmiana formatowania zachodzi automatycznie w całym tekście.
- Jeżeli tekst jest wyjustowany, warto włączyć automatyczne dzielenie wyrazów, aby uniknąć dziur (**Układ → Ustawienia strony → Dzielenie wyrazów → Automatyczne**).
- Praca w trybie recenzji polega na tym, że zmiany wprowadzane w przygotowanym tekście są widoczne. Autor może je później zaakceptować bądź odrzucić. Nie trzeba porównywać wers po wersie tekstu oryginalnego i poprawionego, aby wychwycić poprawki. Ułatwia to i przyśpiesza pracę.
- W tekście w trybie recenzji można też dodawać komentarze. Pozwala to poznać punkt widzenia innej osoby pracującej nad dokumentem. Na komentarz można odpowiedzieć.
- Aby wstawić wzory do tekstu, należy na karcie **Wstawianie** w grupie **Symbole** wybrać opcję **Równanie** i odpowiednio uzupełnić pole do wpisywania równania. W niektórych edytorach równania matematyczne można też wpisywać ręcznie.

## PYTANIA SPRAWDZAJĄCE

1. Jakie są kolejne kroki przygotowywania dokumentu do druku?
2. Co trzeba uwzględnić podczas projektowania layoutu publikacji?
3. Na co należy zwrócić uwagę, gdy tekst jest już wstępnie sformatowany?

## PYTANIA QUIZOWE

1. Co jest efektem modyfikacji stylu?
   **a.** Aktualizacja stylu normalnego.
   **b.** Automatyczna zmiana formatowania w całym tekście.
   **c.** Ponowne zapisanie dokumentu.
   **d.** Sprawdzenie pisowni.

2. Które ze stwierdzeń na temat twardej spacji jest prawdziwe?
   **a.** Wygląda jak zwykła spacja, ale nie pozwala w tym miejscu złamać tekstu.
   **b.** To dwie zwykłe spacje.
   **c.** To przejście do nowego wiersza w ramach tego samego akapitu.
   **d.** Inaczej jest zwana łącznikiem opcjonalnym.
3. Czego **nie** można zrobić w trybie recenzji edytora tekstu?
   **a.** Zaproponować zmian w dokumencie bez edytowania oryginalnego tekstu.
   **b.** Zaproponować zmian formatowania tekstu.
   **c.** Przeprowadzać zmian w dokumencie bez możliwości powrotu do oryginalnego tekstu.
   **d.** Wstawić komentarzy.

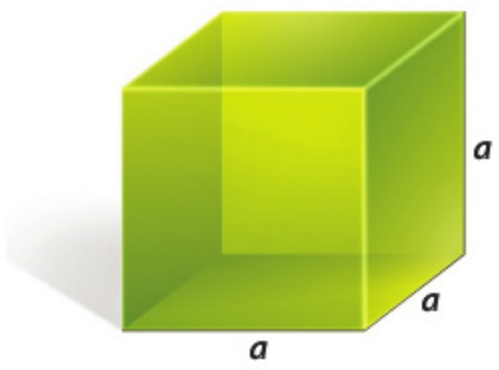
## ZADANIA DODATKOWE

### Zadanie 1. Ulotka na targi edukacyjne

Wykorzystaj szablon przygotowany w ćwiczeniu 1 i przygotuj informacje o szkole na targi edukacyjne.

### Zadanie 2. Pomoc dydaktyczna

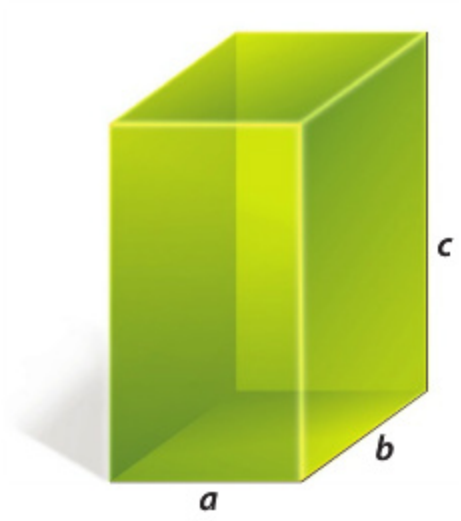
Przygotuj dwukolumnowe tablice zawierające rysunki, treści i wzory dotyczące tematów omawianych obecnie. Po wydrukowaniu dokumentu złóż go na pół.




**Sześcian**

$P_c = 6 \cdot P_p = 6a^2$

$V = a^3$




**Prostopadłościan**

$P_c = 2P_p + P_b = 2(ab + bc + ac)$

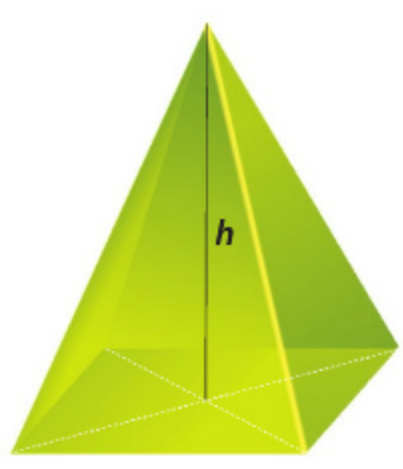
$V = abc$




**Graniastosłup prosty**

$P_c = 2P_p + P_b$

$V = P_p \cdot h$




**Ostrosłup**

$P_c = P_p + P_b$

$V = \frac{1}{3} P_p \cdot h$



# 12. Dokumenty o złożonej strukturze

- Elementy dokumentu o złożonej strukturze
- Przygotowanie wersji papierowej i cyfrowej publikacji
- Opracowywanie spisu treści

## ZADANIE NA START

### Świat czytników

Sprawdź, w jakich formatach są publikowane książki w formie elektronicznej w portalu **https://wolnelektury.pl** oraz w kilku księgarniach internetowych, a następnie wyszukaj informacje na temat tych formatów i przygotuj zestawienie, w którym – oprócz nazwy formatu – zawrzesz jego charakterystykę oraz przeznaczenie.




## ELEMENTY DOKUMENTU O ZŁOŻONEJ STRUKTURZE

Dokument złożony, tzn. składający się z kilku rozdziałów, powinien charakteryzować się pewną strukturą. W przypadku pracy naukowej może ona wyglądać następująco: strona tytułowa, spis treści, wstęp, trzon pracy (rozdziały), spisy tabel i ilustracji, bibliografia, załączniki. Strukturę książki beletrystycznej tworzą z reguły:

- strona tytułowa,
- strona redakcyjna,
- spis treści,
- trzon książki (rozdziały).

Podczas przygotowywania utworu do publikacji warto zadbać o to, aby strukturze logicznej utworu odpowiadała struktura dokumentu. Ułatwia to pracę, np. umożliwia automatyczne przygotowanie spisu treści.

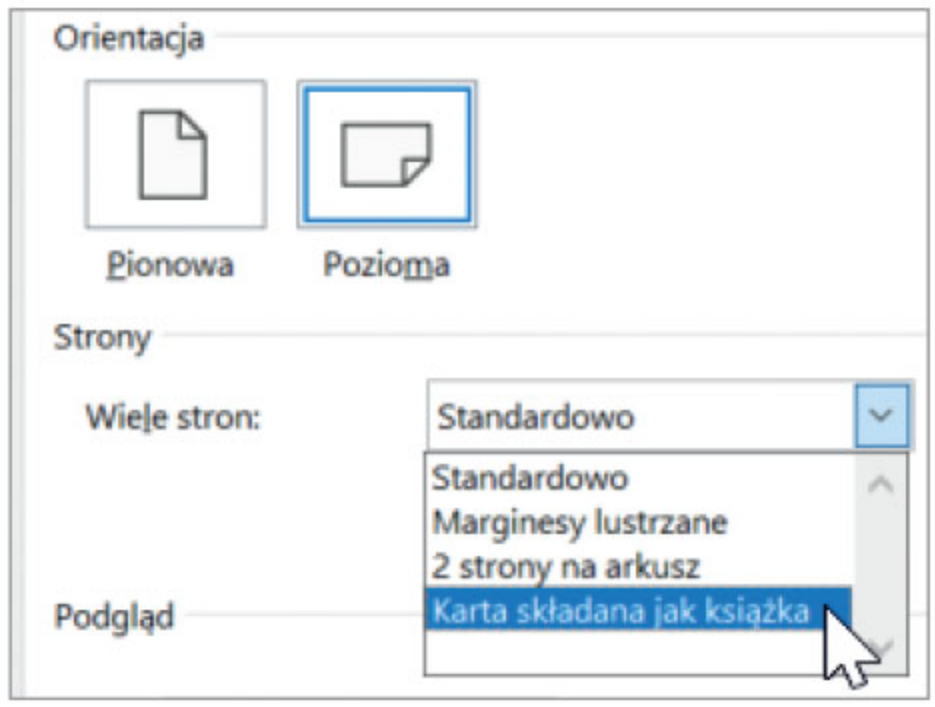
**POSZUKAJ W INTERNECIE**

Porównaj informacje na temat dostępnych na rynku polskich i zagranicznych czytników. Sprawdź, jakie mają parametry, jakie formaty obsługują, jak wypadają podczas testów funkcjonalności.

## UKŁAD DOKUMENTU

Podczas przygotowywania materiału do publikacji w formie książki wiele parametrów można precyzyjnie zdefiniować. Można np. określić wielkość strony, a co za tym idzie – marginesów, a także wygląd czcionek. Ponadto w większej publikacji drukowanej samodzielnie warto korzystać z układu, który zapewnia łączenie stron. Dzięki temu na jednej kartce są drukowane pierwsza i ostatnia strona oraz druga i przedostatnia, potem 3. i 4. oraz 3. i 4. od końca itd. Po wydrukowaniu można wtedy zszyć kartki w środku i powstanie książka.

Aby zaprojektować taką publikację, w programie Word należy wybrać kolejno: **Układ** → **Ustawienia strony** → **Marginesy** → **Marginesy niestandardowe…** → **Wiele stron** → **Karta składana jak książka**. Chociaż formatem papieru jest A4, pojedyncza strona będzie miała rozmiar A5, gdyż dwie strony będą drukowane na jednej kartce.




**Rys. 1.** Ustawienie organizacji tekstu

Oczywiście marginesy należy dostosować do nowego formatu tak, aby nie były ani za duże, ani za małe. Jeśli praca ma być oprawiona, należy ustawić większy margines od strony wewnętrznej.

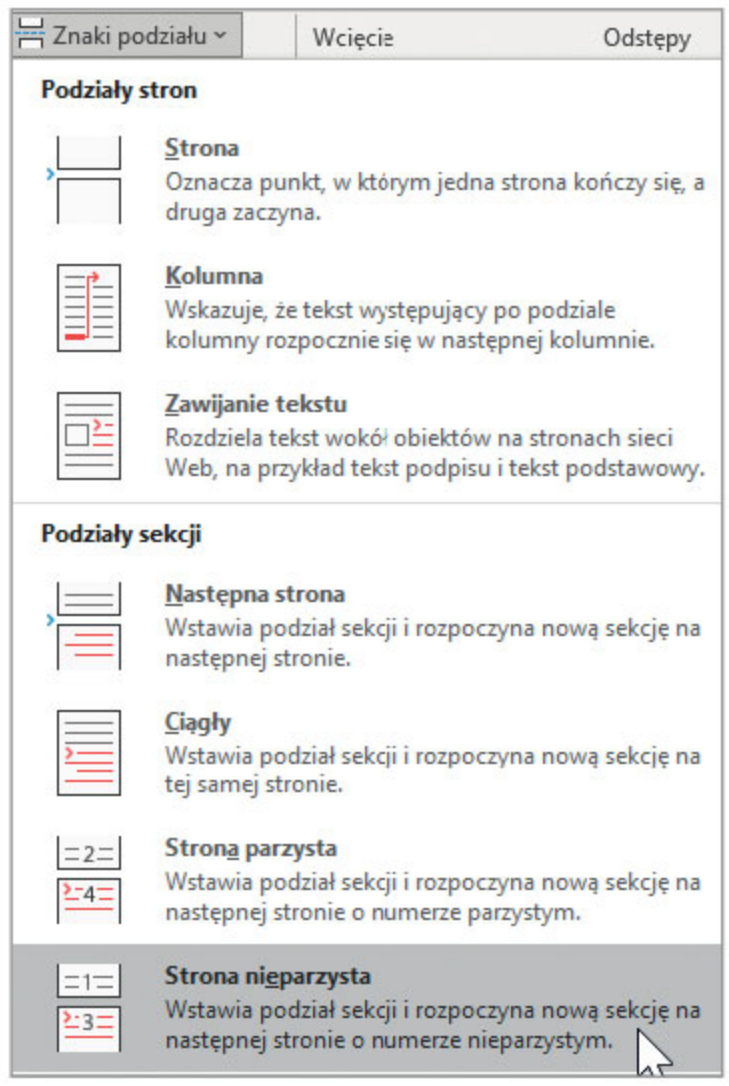
## PODZIAŁ NA SEKCJE

Podział dokumentu na kilka sekcji sprawia, że możliwe staje się różne formatowanie każdej z nich. Inne mogą być np. nagłówki i stopki, liczba kolumn lub nawet orientacja strony.

**ZAPAMIĘTAJ!**

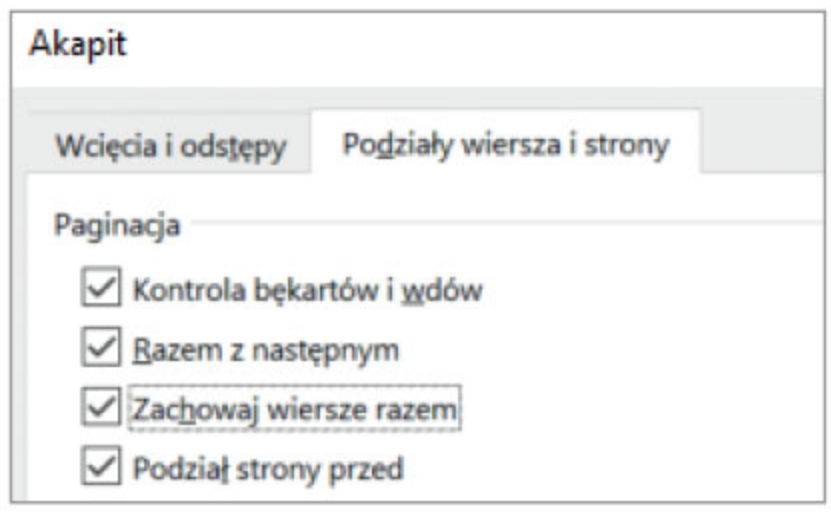
**Sekcje** umożliwiają ustalenie konkretnego układu strony i formatowania różnych części dokumentu. Edytor tekstu traktuje dokument jako pojedynczą sekcję do momentu wstawienia podziału sekcji. **Podział sekcji** steruje układem i formatowaniem poprzedniej sekcji. Po usunięciu podziału tekst uzyskuje formatowanie sekcji następującej po nim.

W książce strony tytułową, redakcyjną i np. z dedykacjami od tłumacza i autora można zostawić bez numeracji, a potem rozpocząć numerację od 1. Aby wstawić znak podziału sekcji, należy ustawić kursor na końcu części wstępnej, a następnie wybrać kolejno: **Układ → Znaki podziału → Strona nieparzysta**. W ten sposób zostanie utworzona nowa sekcja, która rozpocznie się na nowej stronie nieparzystej.




**Rys. 2.** Wstawianie nowej sekcji

Zasadniczo podział na strony następuje automatycznie, ale można też ręcznie wstawić znak podziału strony albo kliknąć **Ustawienia akapitu** w grupie **Akapit** na karcie **Podziały wiersza i strony** i wybrać opcję **Podział strony przed**. Dodatkowo zaznaczenie opcji **Razem z następnym** powoduje, że tytuł rozdziału i pierwszy akapit tekstu danego rozdziału znajdą się na tej samej stronie, czyli na końcu strony nie pozostanie sam tytuł. Aktywna opcja **Kontrola bękartów i wdów** sprawi, że na końcu akapitu lub na początku następnej strony nie zostanie za krótki wiersz.



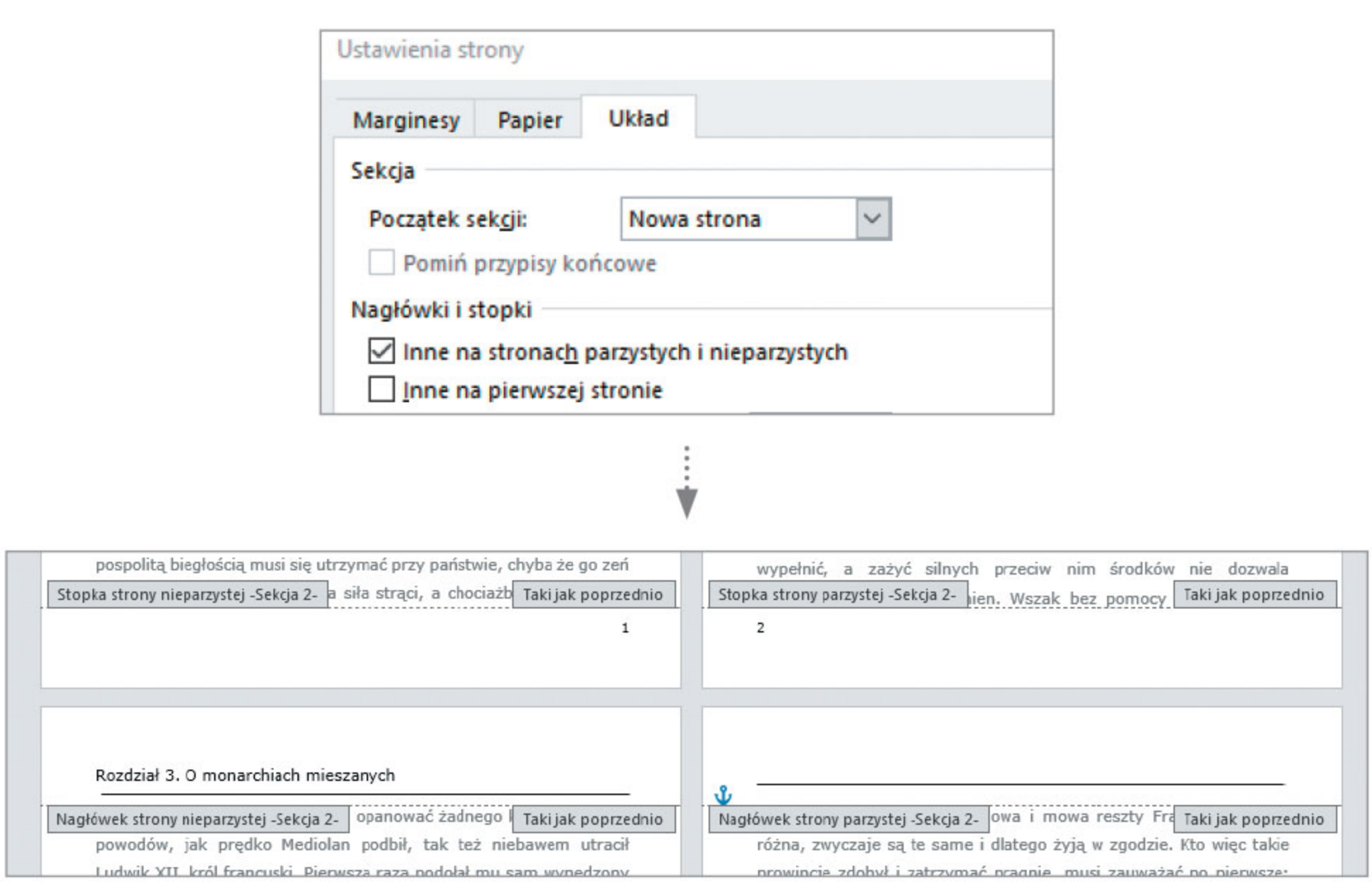

**Rys. 3.** Automatycznie kontrolowane opcje związane z podziałem tekstu na strony

Ostatecznego podziału na strony należy dokonać dopiero po sformatowaniu tekstu, inaczej trzeba będzie wykonywać tę czynność kilkakrotnie.

## NAGŁÓWKI I STOPKI

Nagłówki i stopki w publikacji często bywają inne na stronach parzystych i nieparzystych oraz na pierwszej stronie. Ponadto na pierwszej stronie zwykle nie umieszcza się numeru oraz innych informacji zawartych w nagłówku i stopce.

Aby ustawić numerację stron zawsze po zewnętrznej stronie, należy zaznaczyć na karcie **Układ** w oknie **Ustawienia strony** opcję **Inne na stronach parzystych i nieparzystych**.




**Rys. 4.** Dodawanie opcji umożliwiającej zróżnicowanie nagłówków i stopek na stronach parzystych i nieparzystych

Aby zlikwidować numerację na pierwszej stronie, w sekcji **Nagłówki i stopki** należy wybrać opcję **Inne na pierwszej stronie** i skasować numer strony w nagłówku lub stopce dokumentu na pierwszej stronie.

W edytorze tekstu można wstawić zmienny tekst w nagłówku lub stopce. Aby umieścić tam tytuł rozdziału, trzeba wybrać **Wstawianie** → **Tekst** → **Eksploruj szybkie części** → **Pole**, a następnie w kategorii **Łącza i odsyłacze** wskazać nazwę pola **StyleRef** → **Nagłówek 1**.

**WIEM WIĘCEJ**

**Żywa pagina** to wiersz usytuowany nad kolumną druku lub u dołu kolumny, w którym umieszcza się np. numer strony, tytuł dzieła lub rozdziału, nazwisko autora i tytuł. Zwykle składa się ją pismem o stopniu mniejszym niż tekst podstawowy. **Żywa pagina górna** może być wyśrodkowana, wyrównana do zewnętrznej krawędzi kolumny albo asymetryczna. **Żywą paginę dolną** wyrównuje się do zewnętrznej krawędzi kolumny. Nie wypełnia się całej szerokości kolumny – w razie potrzeby należy uciąć tekst i postawić wielokropek.

## Ćwiczenie 1. Przygotowanie szablonu

Pobierz ze strony **https://wolnelektury.pl** tekst *Traktat o Księciu* w formacie TXT, a następnie opracuj szablon dokumentu, który posłuży ci do przygotowania książki drukowanej.




1. Zdefiniuj odpowiedni układ strony, w tym format i marginesy. W razie potrzeby ustaw też podział na kolumny.
2. Wybierz style dokumentu dla tytułu książki i rozdziału oraz dla tekstu właściwego książki, nagłówka i stopki. Zadbaj o to, aby odpowiadały charakterowi publikacji oraz sprawiały, że książka będzie czytelna.
3. Uzależnij numerację stron i marginesy od tego, czy jest to strona parzysta, czy nieparzysta.

### STRONY TYTUŁOWA I REDAKCYJNA

Strona tytułowa książki zwykle zawiera: tytuł w pełnym brzmieniu, imię i nazwisko autora, imię i nazwisko tłumacza, nazwę i logo wydawnictwa, rok. Na stronie redakcyjnej umieszcza się komplet informacji wydawniczych i drukarskich, w tym nazwiska współtwórców książki niewymienione na stronie tytułowej, notkę o prawach autorskich, ISBN i numer wydania oraz wszelkie inne informacje techniczno-wydawnicze.

## Ćwiczenie 2. Opracowanie stron tytułowej i redakcyjnej

Przygotuj strony tytułową i redakcyjną książki z ćwiczenia 1.

1. Zamieść wszystkie potrzebne informacje.
2. Zadbaj o to, aby formatowanie służyło czytelnemu przedstawieniu treści.

## Ćwiczenie 3. Opracowanie tekstu

Wklej do szablonu z ćwiczenia 1 pobrany tekst powieści oraz stronę tytułową i stronę redakcyjną przygotowane w ramach ćwiczenia 2, a następnie opracuj materiał.

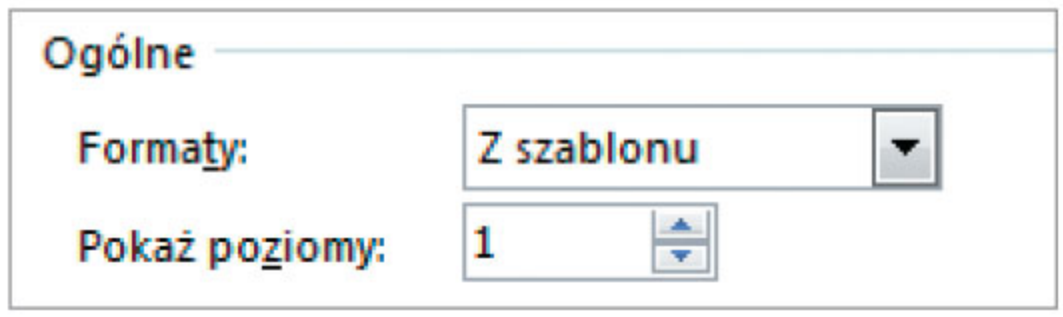
1. Nadaj poszczególnym elementom odpowiednie style.
2. Wyeliminuj ewentualne błędy typograficzne, zadbaj o to, aby tekst był poprawnie i przejrzyście sformatowany.

**W PRAKTYCE**

Aby wstawić przypis, na karcie **Odwołania** w grupie **Przypisy dolne** należy wybrać opcję **Wstaw przypis dolny** lub **Wstaw przypis końcowy**.

## SPIS TREŚCI

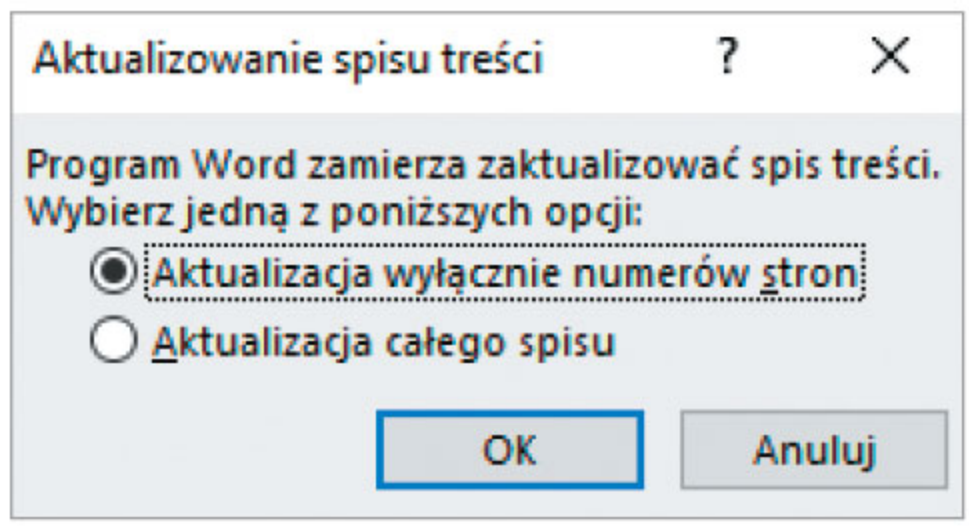
Po przypisaniu stylów tytułom rozdziałów można przystąpić do automatycznego przygotowania spisu treści. W tym celu na karcie **Odwołania** należy wybrać **Spis treści** oraz konkretny jego rodzaj. Po wybraniu opcji **Niestandardowy spis treści** możliwe staje się określenie dodatkowych parametrów, np. liczby poziomów do uwzględnienia w spisie treści.




**Rys. 5.** Ustawienie, zgodnie z którym w spisie treści znajdą się tylko nagłówki poziomu 1

W przypadku tekstów naukowych lub popularnonaukowych należy dodatkowo sporządzić spis ilustracji oraz bibliografię.

Jeśli w dokumencie wprowadzono większe zmiany, warto zaktualizować spis treści. W tym celu należy kliknąć prawym przyciskiem myszy w dowolnym miejscu tego spisu, a następnie wybrać opcję **Aktualizuj pole** i jedną z dwóch opcji: **Aktualizacja wyłącznie numerów stron** lub **Aktualizacja całego spisu**. Wybór zależy od tego, czy zmiany dotyczyły tylko podziału na strony, czy też dodano nowe rozdziały.




**Rys. 6.** Aktualizacja spisu treści

### Ćwiczenie 4. Przygotowanie publikacji do druku

Dopracuj książkę przygotowaną w ćwiczeniu 3 tak, aby była gotowa do druku.

1. Wstaw okładkę przygotowaną podczas lekcji 9 oraz ozdobniki graficzne – wykaż się inicjatywą twórczą.
2. Dołącz spis treści na początku lub na końcu publikacji.

### Ćwiczenie 5. Ocena publikacji

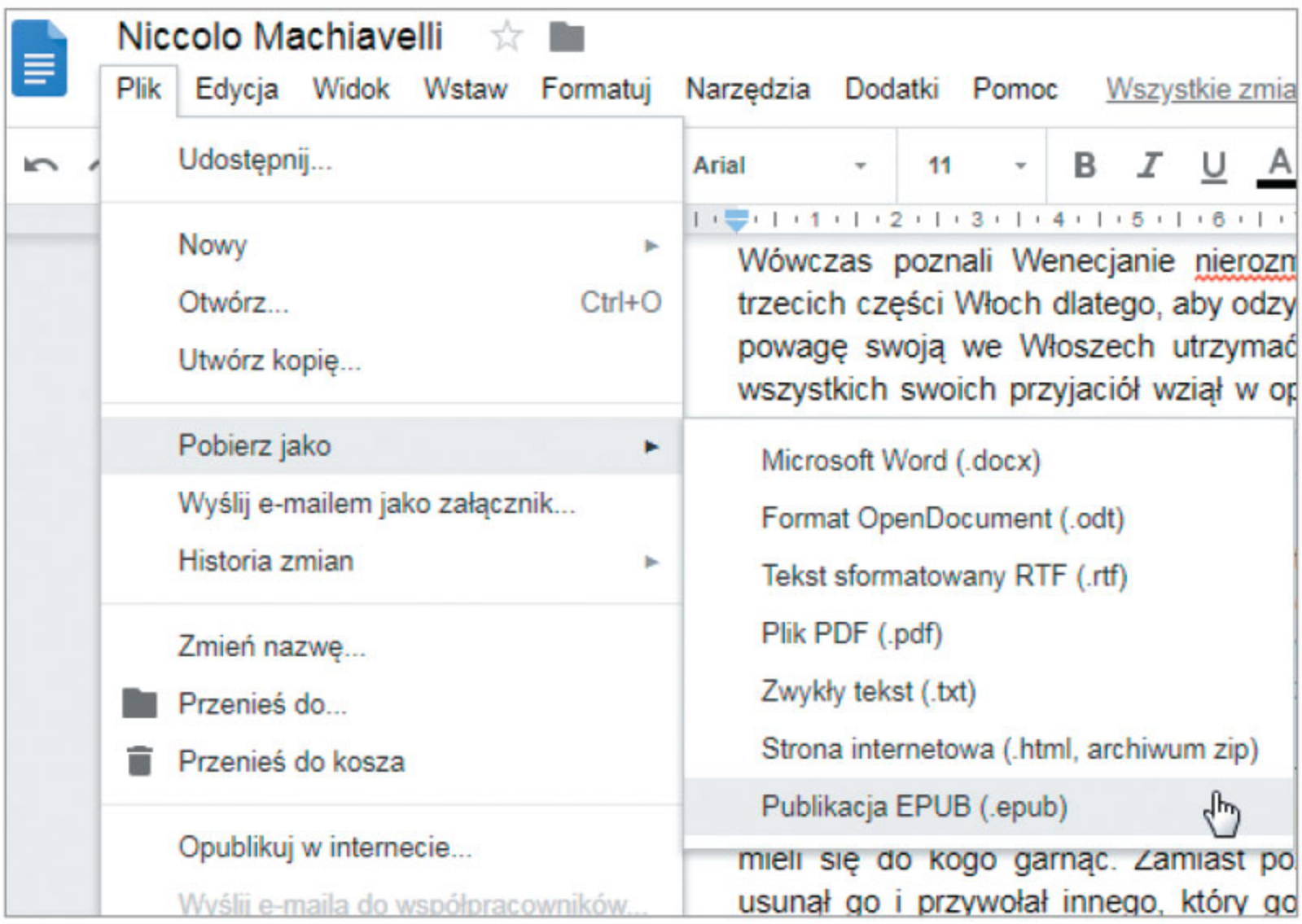
Przygotujcie wspólnie kartę oceny publikacji, zawierającą co najmniej 15 pytań dotyczących m.in. układu strony, formatowania akapitów i poprawności formatowania na poziomie znaków, a następnie w drodze losowania wybierzcie trzy prace do oceny na forum klasy. Poddajcie je konstruktywnej krytyce – starajcie się uzasadnić każdą ocenę oraz zasugerować sposób rozwiązania problemu.

**W PRAKTYCE**

Oprócz automatycznego spisu treści można też wygenerować **spis tabel** czy **rysunków**. Trzeba w tym celu wcześniej przygotować odpowiednie odwołania. Pod każdym rysunkiem lub tabelą trzeba wstawić podpis (**Odwołania** → **Wstaw podpis**), ponadto dla tabeli wybierać etykietę **tabela**. Gdy już każdy element jest opatrzony podpisem, wystarczy przejść kursorem w miejsce, gdzie ma być wstawiony spis, i wybrać opcję **Wstaw spis ilustracji**.

## PRZYGOTOWANIE E-BOOKA

E-book to książka elektroniczna przeznaczona do czytania np. na komputerze, czytniku czy smartfonie za pomocą odpowiedniego oprogramowania. Obecnie najpopularniejsze formaty e-booków to EPUB oraz MOBI. Niektóre edytory tekstowe oferują opcję eksportu do wymaganego formatu – taką funkcję można znaleźć m.in. w LibreOffice od wersji 6.0 oraz w Dokumentach Google. Sprawdź, czy ma ją twój edytor.

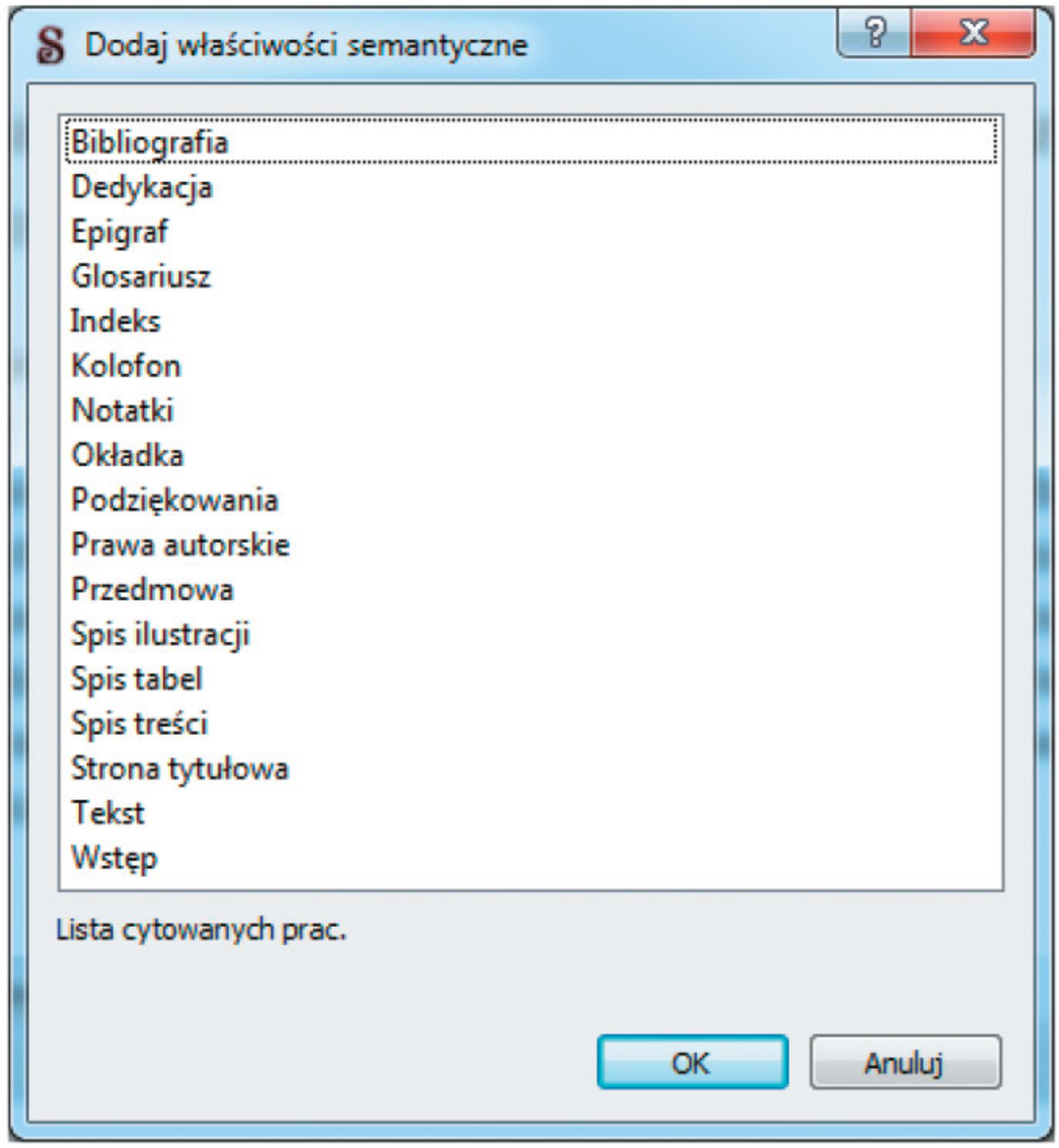



**Rys. 7.** Eksport *Traktatu o Księciu* Niccola Machiavellego do formatu EPUB w Dokumentach Google

E-book można również przygotować za pomocą programu przeznaczonego specjalnie do tego celu. Pozwala to uwzględnić kilka dodatkowych czynników (np. różnej wielkości ekrany czytników, różne możliwości wyświetlania grafik, inne zasoby narzędzi do formatowania) i zastosować rozwiązania, dzięki którym dostęp do treści będzie miało szersze grono odbiorców.

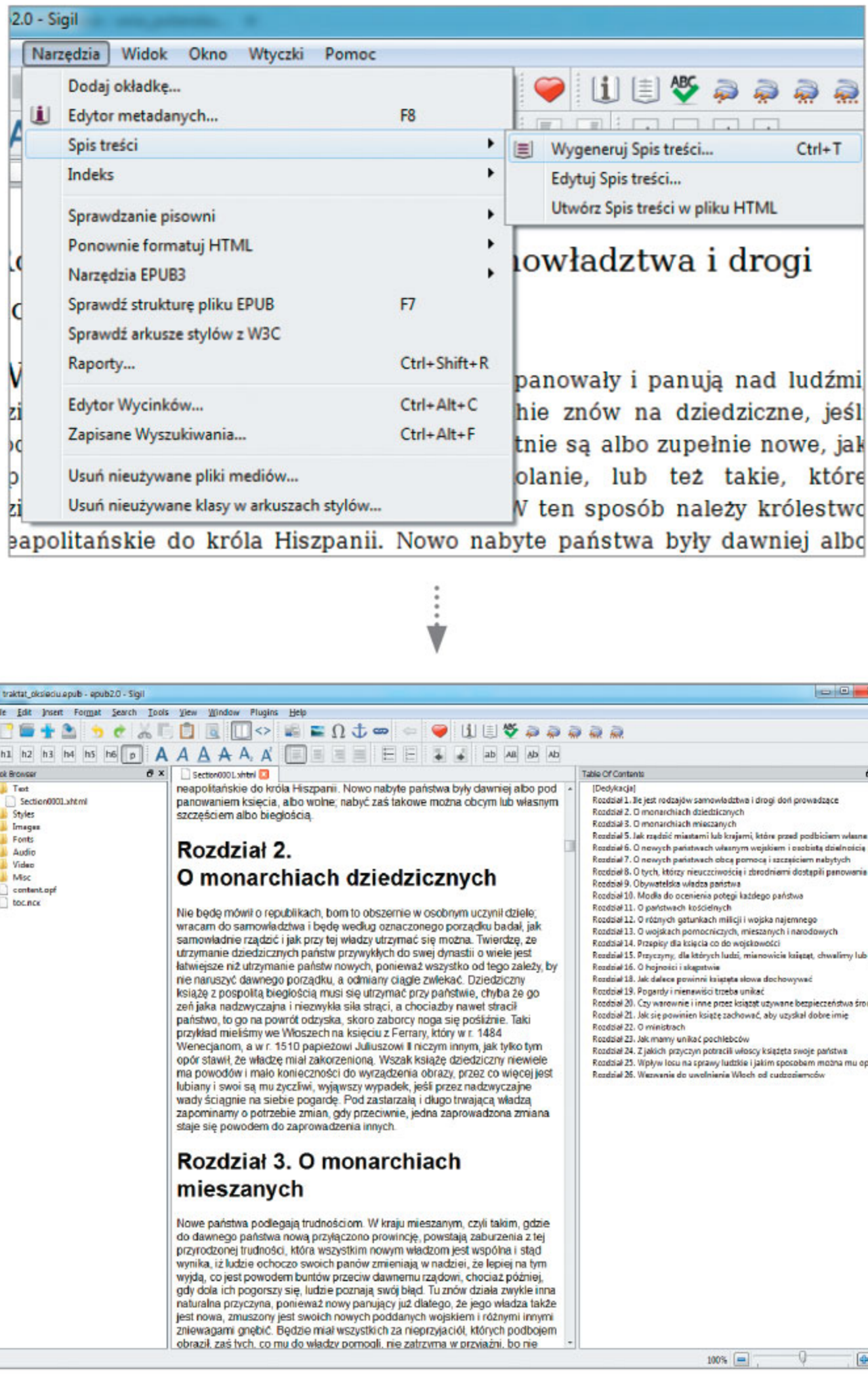
Aplikacja Sigil (**https://github.com/Sigil-Ebook/Sigil**) obsługuje formaty ePUB2, ePUB3 oraz MOBI. Pozwala ona na formatowanie tekstu (np. stosowanie pogrubień, kursywy czy podkreśleń, wstawianie list punktowanych i numerowanych, ilustracji oraz wcięć), generowanie spisu treści, sprawdzanie pisowni za pomocą domyślnych i konfigurowanych przez użytkownika słowników oraz import i eksport plików. Podstawowe czynności w tej aplikacji można wykonać w trybie WYSIWYG (*What You See Is What You Get* – co widzisz, to otrzymujesz), lecz aby w pełni korzystać z edytora, warto znać język HTML.

Okładce, stronie tytułowej, bibliografii i tym podobnym elementom można nadać odpowiednią semantykę – dzięki temu zabiegowi okładka w każdym czytniku zostanie wyświetlona jako okładka, a nie np. wstęp czy przedmowa. Aby wybrać tę opcję, należy kliknąć prawym przyciskiem myszy w spis zawartości książki i wskazać w menu **Dodaj semantykę**. Potem wystarczy zaznaczyć odpowiednią pozycję, np. **Okładka**.




**Rys. 8.** Dodawanie właściwości semantycznych w Sigilu

Aby dodać spis treści, należy wybrać na pasku menu **Narzędzia → Spis treści → Wygeneruj Spis treści**.




**Rys. 9.** Dodawanie spisu treści do książki *Traktat o Księciu* w Sigilu

Na zakończenie pracy nad e-bookiem warto uzupełnić **metadane**, np. tytuł tekstu, jego autora czy zastosowany język. Należy wybrać **Narzędzia → Edytor metadanych…**

### Ćwiczenie 6. Przygotowanie publikacji elektronicznej

Przygotuj drugą wersję *Traktatu o Księciu* na potrzeby publikacji w formie elektronicznej. Nie zapomnij o spisie treści. Wyeksportuj dokument do formatu akceptowanego przez e-book, którym dysponujesz. Jeśli to możliwe, przetestuj swój projekt na kilku urządzeniach.

## PODSUMOWANIE

- Podczas przygotowywania dokumentu do publikacji należy wyodrębnić poszczególne jego elementy tak, aby struktura była logiczna i przejrzysta.
- Przygotowywanie tekstu do druku i publikacji w formie elektronicznej ma wiele cech wspólnych, warto jednak zwrócić uwagę, że na potrzeby druku można określić docelowy rozmiar strony, zbiór czcionek, którymi dysponujemy, itp. Dokumenty elektroniczne mogą natomiast być wyświetlane na urządzeniach o różnych parametrach.
- Spisy treści, tabel i rysunków można wygenerować automatycznie, jeśli dokument ma nadaną odpowiednią strukturę, czyli rozdziały i podrozdziały, albo odpowiednio podpisy pod tabelami i rysunkami są zdefiniowane za pomocą stylów.

## PYTANIA SPRAWDZAJĄCE

1. Jakie są kolejne etapy przygotowywania książki do druku?
2. Na co trzeba zwrócić uwagę przy projektowaniu książki oraz e-booka?
3. Jakie są typowe błędy typograficzne i jak się ich ustrzec?

## PYTANIA QUIZOWE

1. W jakim celu stosuje się sekcje w dokumencie tekstowym?
   **a.** W celu odróżnienia części ważniejszej od mniej ważnej.
   **b.** Aby ustawić różne układy strony w jednym dokumencie.
   **c.** Aby podzielić tekst na strony.
   **d.** Aby ponumerować strony.
2. Co jest konieczne do utworzenia automatycznego spisu treści?
   **a.** Ponumerowanie stron.
   **b.** Wyróżnienie tytułów rozdziałów pogrubieniem.
   **c.** Zapisanie pliku w formacie PDF.
   **d.** Stosowanie stylów z nagłówkami.
3. Co trzeba zrobić, żeby zlikwidować numerację na pierwszej stronie?
   **a.** Ustawić rozpoczęcie numeracji stron od –1.
   **b.** Na karcie **Układ** w sekcji **Nagłówki i stopki** zaznaczyć opcję **Inne na pierwszej stronie**.
   **c.** Ustawić rzymskie znaki przy ustalaniu sposobu numerowania.
   **d.** Oznaczyć stronę w innej orientacji, np. poziomej.

## ZADANIE DODATKOWE

### Zadanie 1. *Książę* ilustrowany

Dodaj do pliku z ćwiczenia 4 pasujące ilustracje wraz z podpisami oraz spis ilustracji.



# 13. Korespondencja seryjna

- Etapy tworzenia korespondencji seryjnej
- Przygotowanie danych i opracowanie wzorca
- Generowanie serii dokumentów

## ZADANIE NA START

### Spersonalizowany e-mail

**Cześć, Zuzanna Kwiatkowska**

W przyszłym miesiącu zamkniemy jeden z naszych sklepów w Twoim regionie.
Z tego powodu ogłaszamy marzec miesiącem likwidacji zapasów.

Dziś Twój e-mail: **zuzxxana@gmail.com** został wybrany, abyś mogła otrzymać dowolną parę sportowych butów za 14 zł (zamiast standardowej ceny 480 zł).

**Kliknij tutaj po więcej informacji**

Z poważaniem,
obsługa klienta

Poszukaj w skrzynce pocztowej podobnego e-maila zawierającego twoje dane osobowe. Wyodrębnij te fragmenty wiadomości, które są wysyłane do każdego, oraz te przeznaczone tylko dla ciebie.

## TWORZENIE KORESPONDENCJI SERYJNEJ

Czasami trzeba się skomunikować z większą liczbą osób. Wówczas konieczne jest przygotowanie dużej liczby etykiet adresowych, listów, e-maili, raportów, zaświadczeń, kopert, zaproszeń i innych dokumentów różniących się tylko kilkoma szczegółami. Realizacja takiego zadania w przypadku tworzenia każdego pliku osobno wymagałaby ogromnych nakładów pracy i czasu, zwłaszcza gdyby się okazało, że na końcowym etapie trzeba wprowadzić jeszcze jakieś poprawki. Dlatego warto do tego celu wykorzystać korespondencję seryjną, dzięki której w kilka minut można wygenerować tyle dokumentów, ilu jest adresatów w bazie.

**ZAPAMIĘTAJ!**

**Korespondencja seryjna** to funkcjonalność pozwalająca na przygotowanie w edytorze tekstu jednakowych dokumentów, różniących się zawartością kluczowych pól, którą pobiera się z bazy danych.

Aby utworzyć dokument do korespondencji seryjnej, należy przygotować bazę danych oraz szablon z tekstem (dokument wzorcowy) i odwołaniami do bazy danych, a następnie je scalić. Powstanie wówczas gotowy dokument, w którym puste pola zostaną wypełnione przygotowanymi wcześniej danymi z bazy.

Zapraszam «imię» «nazwisko» na wieczór filmowy, czyli klasowe oglądanie filmu *Upiorne wakacje*.
Spotykamy się w piątek o godzinie 19.00 u mnie w domu.
Do zobaczenia,
Ignacy

| | A | B |
|---|---|---|
| 1 | **Imię** | **Nazwisko** |
| 2 | Marysię | Nowak |
| 3 | Antka | Kowalskiego |
| 4 | Wojtka | Polaka |
| 5 | Helenę | Ziembińską |

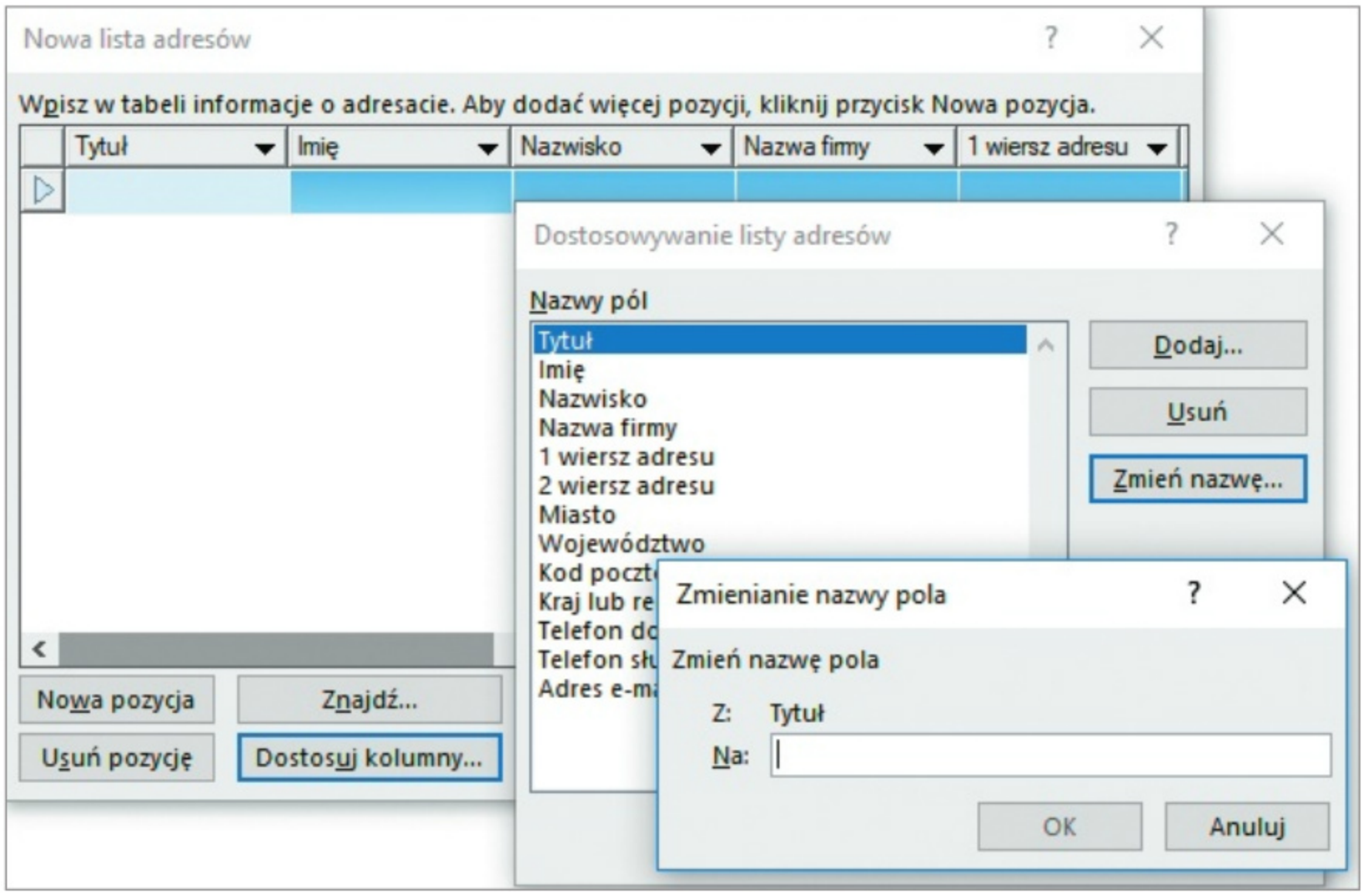
Zapraszam Marysię Nowak na wieczór filmowy, czyli klasowe oglądanie filmu *Upiorne wakacje*.
Spotykamy się w piątek o godzinie 19.00 u mnie w domu.
Do zobaczenia,
Ignacy

Zapraszam Antka Kowalskiego na wieczór filmowy, czyli klasowe oglądanie filmu *Upiorne wakacje*.
Spotykamy się w piątek o godzinie 19.00 u mnie w domu.
Do zobaczenia,
Ignacy

**Rys. 1.** Szablon zaproszenia, baza danych i dokument uzupełniony danymi pobranymi z bazy

## BAZA DANYCH

Mniejsza liczba danych oznacza, że tabelę źródłową można przygotować w edytorze tekstu podczas opracowywania dokumentu wzorcowego. W Wordzie należy wybrać kartę **Korespondencja**, rozwinąć listę **Wybierz adresatów**, wybrać opcję **Wpisz nową listę…**, a następnie odpowiednio zmodyfikować i uzupełnić wyświetloną tabelę. W edytorze Writer poprowadzi cię **Kreator korespondencji seryjnej**, który znajdziesz w menu **Narzędzia**.




**Rys. 2.** Okno umożliwiające tworzenie źródła danych bezpośrednio w edytorze Word

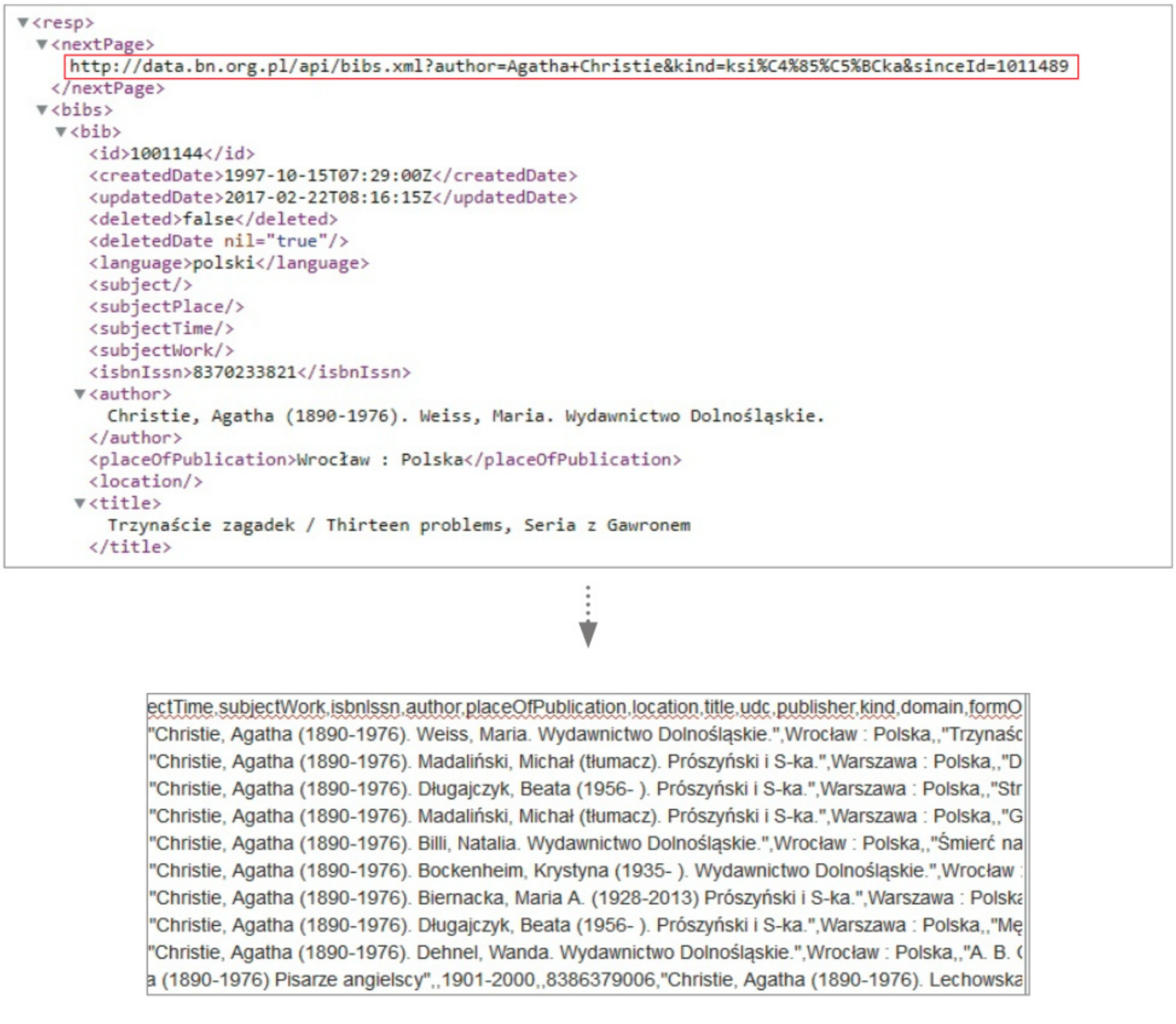
W przypadku większej liczby danych tabelę warto opracować w arkuszu kalkulacyjnym.

- Jeśli bazę danych stanowi lista kontaktów programu pocztowego, użytkownicy aplikacji Microsoft Word i Microsoft Outlook mogą wybrać odpowiednie kontakty z poziomu wstążki (na karcie **Korespondencja** należy rozwinąć listę **Wybierz adresatów** i zaznaczyć opcję **Wybierz z kontaktów programu Outlook**).
- Osoby korzystające z innych aplikacji pocztowych lub innych edytorów tekstu muszą eksportować listę kontaktów w formacie XML, który został stworzony właśnie w celu łatwego przenoszenia i przechowywania danych. Plik trzeba następnie przekonwertować na format CSV, w którym wartości są oddzielane przecinkami. Do popularnych konwerterów plików XML na CSV należą m.in. XML to CSV Conversion Tool, Moor XML to CSV Converter czy Open Refine. Dane z pliku CSV należy następnie wczytać do arkusza kalkulacyjnego.

**WIEM WIĘCEJ**

**Język XML** (*eXtensible Markup Language* – rozszerzalny język znaczników) jest niezależny od platformy. Dlatego dokumenty zapisane jako XML tak łatwo wymienia się między różnymi systemami. Jest to **język do tworzenia innych języków**, a nie dokumentów. I dopiero te języki służą do budowania dokumentów w internecie. Te języki są nazywane aplikacjami XML, najbardziej znaną jest XHTML.






**Rys. 3.** Wyniki wyszukiwania listy książek, których autorką jest Agatha Christie, w formacie XML i po przekonwertowaniu na CSV

## Ćwiczenie 1. Przygotowanie bazy danych

Pobierz listę sześciu kryminałów Agathy Christie z serwisu API Biblioteki Narodowej (**http://data.bn.org.pl**), a następnie opracuj ją w arkuszu kalkulacyjnym. Każdy rekord zilustruj obrazkiem okładki.

1. Otwórz serwis API i wybierz kartę **Rekordy bibliograficzne**.
2. Ustaw kryteria wyszukiwania – określ typ materiału (książka), autora (Agatha Christie), limit wyników (6) i format (XML). W razie potrzeby rozwiń listę **Nowe pole** i za pomocą zielonego przycisku dodaj do formularza potrzebne kategorie, np. język.
3. Zapisz plik XML na dysku w folderze przeznaczonym na ten projekt.
4. Przekonwertuj plik XML na format CSV i wczytaj go do arkusza kalkulacyjnego.
   **Dane → Pobierz dane → Z pliku → Z pliku tekstowego/CSV**

5. Uporządkuj dane i przekopiuj do nowego arkusza tylko kolumny wykorzystywane do korespondencji seryjnej – **Autor** i **Tytuł**. Gdy masz do czynienia z książkami jednego autora, w tabeli możesz pominąć kolumnę z nagłówkiem **Autor** – każdy rekord ma tutaj tę samą wartość.

| | A |
|---|---|
| 1 | **Tytuł** |
| 2 | Trzynaście zagadek |
| 3 | Dlaczego nie Evans? |
| 4 | Strzały w Stonygates |
| 5 | Godzina zero |
| 6 | Śmierć na Nilu |
| 7 | Kot wśród gołębi |

6. Dodaj kolumnę **Okładka**, w której umieścisz nazwy plików graficznych, i zapisz plik arkusza (np. **dane.xlsx**) w tym samym folderze co plik XML.

7. Przygotuj odpowiednie pliki JPG i dodaj do folderu.

8. Wpisz nazwy plików graficznych.

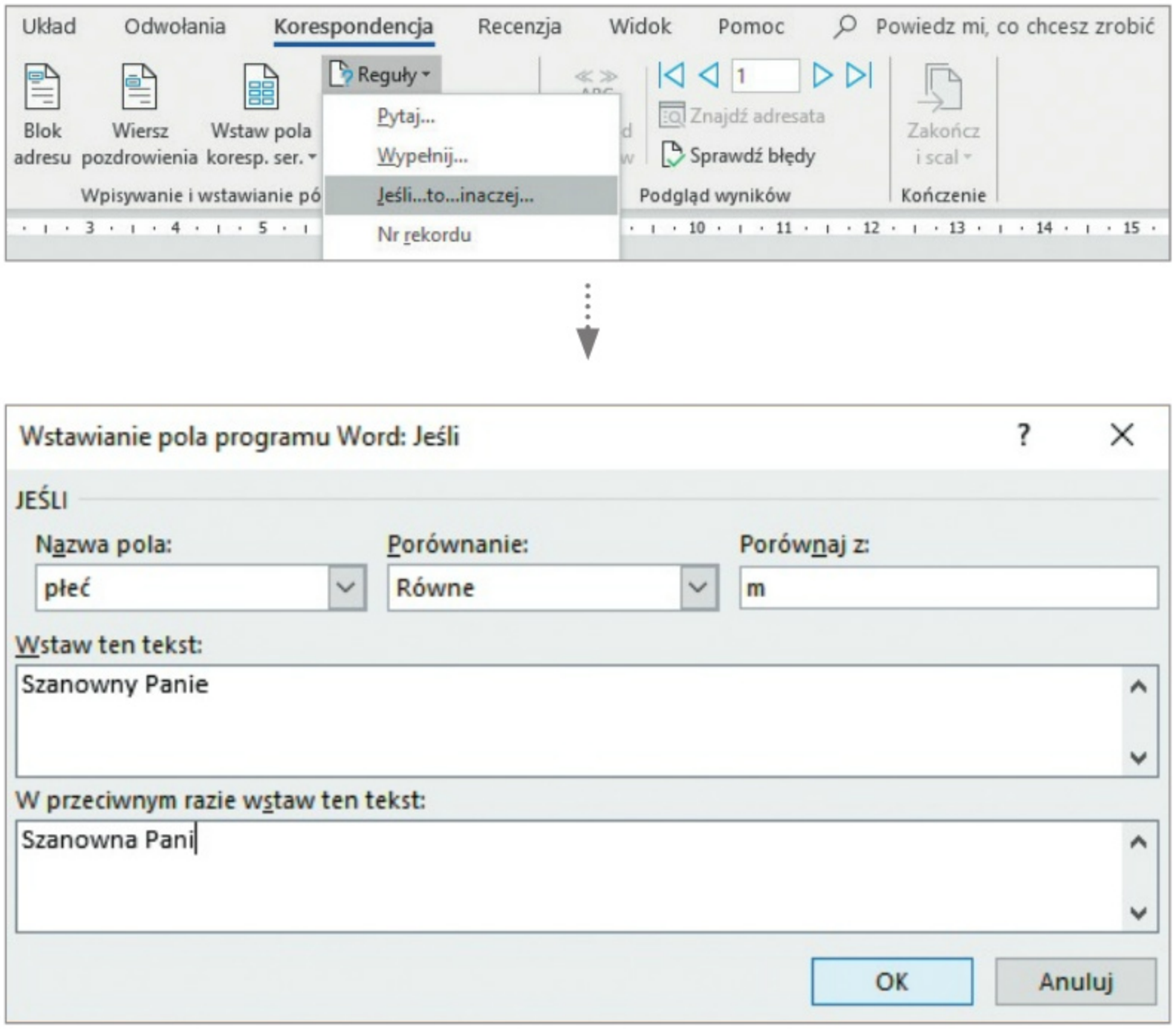
| | A | B |
|---|---|---|
| 1 | **Tytuł** | **Okładka** |
| 2 | Trzynaście zagadek | ach1.jpg |
| 3 | Dlaczego nie Evans? | ach2.jpg |
| 4 | Strzały w Stonygates | ach3.jpg |
| 5 | Godzina zero | ach4.jpg |
| 6 | Śmierć na Nilu | ach5.jpg |
| 7 | Kot wśród gołębi | ach6.jpg |

## SZABLON DOKUMENTU

Aby rozpocząć tworzenie szablonu (dokumentu wzorcowego) w programie Microsoft Word, należy wybrać kartę **Korespondencja**, rozwinąć listę **Rozpocznij koresp. ser.** i wybrać typ dokumentu (listy, wiadomości e-mail, koperty, etykiety, katalogi, zwykłe dokumenty) lub opcję **Kreator korespondencji seryjnej krok po kroku…**, który zgodnie ze swoją nazwą prowadzi użytkownika przez cały proces tworzenia i generowania dokumentu wzorcowego.

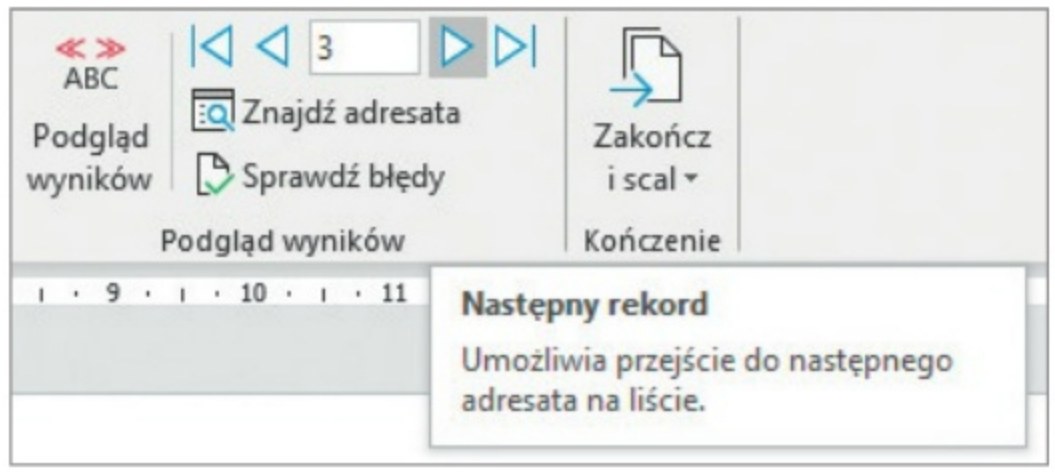
W przypadku listów można skorzystać z gotowych szablonów lub istniejącego już dokumentu, w przypadku kopert czy etykiet – zmienić układ dokumentu. Następnie należy wybrać adresatów z bazy danych i przygotować dokument według własnego pomysłu, w tym odpowiednio wstawić pola korespondencji seryjnej.

Narzędzie do korespondencji seryjnej pozwala na konfigurowanie reguł – stosowanie pól warunkowych, za pomocą których można np. łatwo zdefiniować zwrot inny dla kobiety, a inny dla mężczyzny. W bazie danych trzeba wówczas umieścić kolumnę zatytułowaną np. **płeć** i odpowiednio ją uzupełnić (wystarczą litery *m* na określenie mężczyzny i *k* – na określenie kobiety), a następnie w edytorze tekstu na karcie **Korespondencja** w grupie **Wpisywanie i wstawianie pól** rozwinąć listę **Reguły** i wybrać **Jeśli…to…inaczej….**




**Rys. 4.** Wstawianie reguły **Jeśli…to…inaczej…**

Aby upewnić się, że informacje w gotowym szablonie wyświetlają się poprawnie, warto na koniec pracy wybrać opcję **Podgląd wyników**. Przyciski **Następny rekord** i **Poprzedni rekord** pozwalają poruszać się między kolejnymi rekordami.




**Rys. 5.** Podgląd wyników

## Ćwiczenie 2. Przygotowanie szablonu

Przygotuj dokument wzorcowy, w którym znajdą się następujące dane: imię i nazwisko autora książki oraz tytuł i obrazek okładki. Dane pobierz z bazy przygotowanej w ćwiczeniu 1.

1. W edytorze tekstu utwórz pusty dokument i zapisz go na dysku w tym samym folderze, w którym znajdują się plik arkusza i pliki graficzne.

2. Rozpocznij tworzenie korespondencji seryjnej. Wskaż plik z danymi.
   **Korespondencja → Rozpocznij koresp. ser. → Listy**
   **Wybierz adresatów → Użyj istniejącej listy...**

3. W przypadku książek jednego autora edycję dokumentu rozpocznij od uzupełnienia imienia i nazwiska. Potem w nowym wierszu wstaw pole korespondencji seryjnej.
   **Wstaw pola koresp. ser. → Tytuł**

4. Dodaj do dokumentu pliki graficzne – zastosuj skrót klawiaturowy **Alt+F9**, aby przełączyć się między wyświetlaniem kodów wszystkich pól a wyświetlaniem ich wyników.

   **«Tytuł»** ← **Alt+F9** → **{ MERGEFIELD Tytuł }**

5. Wstaw w nowym wierszu puste pole w nawiasach klamrowych (nie można go wstawić przez wpisanie znaków z klawiatury) – skorzystaj z kombinacji klawiszy **Ctrl+F9**.

6. Do wstawienia obrazka w dokumencie potrzebny jest kod pola **INCLUDEPICTURE** – wpisz kod w środku wstawionych nawiasów, wstaw spację, a następnie podaj w cudzysłowie ścieżkę do pliku z grafiką.

   **Agatha Christie**
   **{ MERGEFIELD tytuł } { INCLUDEPICTURE "{ MERGEFIELD Okładka }" }**

7. Wróć do standardowego widoku za pomocą klawiszy **Alt+F9** – kod pola zniknie, pozostanie tylko wstawione wcześniej pole **Tytuł** (okładki będą widoczne dopiero w scalonym dokumencie).

8. Sformatuj wpisany tekst i pole **Tytuł**.

### SCALANIE DOKUMENTU

Aby scalić szablon i bazę danych, na karcie **Korespondencja** w grupie **Kończenie** należy rozwinąć listę **Zakończ i scal**, a potem w zależności od potrzeb wydrukować lub wysłać przygotowany dokument albo wybrać opcję **Edytuj poszczególne dokumenty**. Powoduje ona utworzenie nowego pliku ze scalonymi dokumentami, który można zapisać i edytować.

## Ćwiczenie 3. Scalenie bazy danych i szablonu

Przygotuj scalony dokument do prezentacji lub wydruku na spotkanie klubu miłośników twórczości Agathy Christie. Sprawdź jego poprawność.

1. Scal dokument tekstowy z dokumentem zawierającym źródła danych, aby otrzymać nowy dokument, który będzie miał tyle stron, ile jest rekordów w bazie.
   **Korespondencja → Zakończ i scal → Edytuj poszczególne dokumenty → OK**
2. Wstaw grafiki – zaznacz cały dokument i zaktualizuj zaznaczone pola klawiszem **F9**.
3. Gotowy do wydruku scalony dokument zapisz jako osobny plik na dysku.

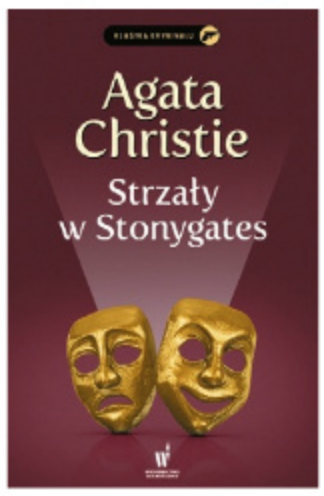
**Agatha Christie**
**Trzynaście zagadek**



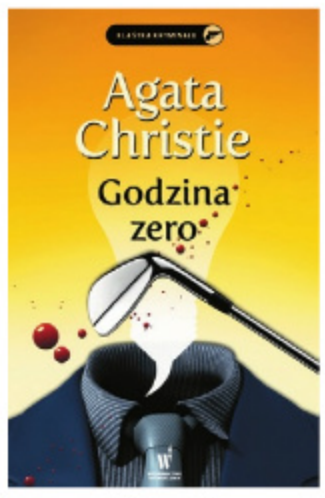

**Agatha Christie**
**Dlaczego nie Evans?**




**Agatha Christie**
**Strzały w Stonygates**




**Agatha Christie**
**Godzina zero**




**Agatha Christie**
**Śmierć na Nilu**




**Agatha Christie**
**Kot wśród gołębi**




### PODSUMOWANIE

- Korespondencja seryjna umożliwia opracowywanie wielu dokumentów naraz, m.in. zbiorcze adresowanie kopert, przygotowywanie etykiet, pisanie listów i e-maili. W przypadku dużej liczby rekordów skutkuje to wymiernym skróceniem czasu pracy i pozwala uniknąć wielu błędów.
- Aby utworzyć korespondencję seryjną, należy przygotować bazę danych oraz szablon z tekstem i odwołaniami do bazy danych, a następnie scalić oba dokumenty.

- Dane do korespondencji seryjnej można pobierać z różnych źródeł, np. z arkusza kalkulacyjnego, programu pocztowego oraz edytora tekstu.
- Kreator korespondencji seryjnej prowadzi użytkownika przez cały proces tworzenia dokumentu wzorcowego oraz scalania go z bazą danych.
- Podczas konfigurowania korespondencji seryjnej można dodawać reguły pozwalające zmienić niektóre fragmenty treści w zależności od podanego warunku, np. dostosować pozdrowienie do płci.

## PYTANIA SPRAWDZAJĄCE

1. Kiedy warto stosować korespondencję seryjną?
2. Do czego służy scalanie w korespondencji seryjnej?
3. Co trzeba zrobić, żeby przy tworzeniu listy za pomocą funkcji korespondencji seryjnej uwzględnić płeć adresata?

## PYTANIA QUIZOWE

1. Co należy przygotować podczas tworzenia korespondencji seryjnej?
   **a.** Bazę danych nadawców oraz szablon z tekstem.
   **b.** Drukarkę sieciową.
   **c.** Bazę danych korespondentów oraz szablon z tekstem.
   **d.** Dokument wzorcowy w edytorze i PDF-ie.
2. Co jest efektem scalania dokumentu podczas korespondencji seryjnej?
   **a.** Nowy dokument, który można edytować.
   **b.** Nowy dokument, którego nie można edytować.
   **c.** Nowy dokument, którego nie można zapisać.
   **d.** Nowy dokument, którego nie można drukować.
3. Chcesz przygotować dyplomy w formacie A4. Co możesz wybrać, aby rozpocząć korespondencję seryjną?
   **a.** Etykiety.
   **b.** Koperty.
   **c.** Listy.
   **d.** Wiadomości e-mail.

## ZADANIE DODATKOWE

### Zadanie 1. Uzupełnianie bazy

Rozbuduj opracowany podczas lekcji dokument – dodaj krótki opis książki oraz ISBN, umieść imię i nazwisko autora na etykiecie. Wyślij tak przygotowany dokument za pomocą korespondencji seryjnej do wszystkich członków klubu miłośników twórczości Agathy Christie z twojej klasy.




# ALGORYTMIKA I PROGRAMOWANIE W PYTHONIE

Programowanie, które łączy myślenie kreatywne, abstrakcyjne, logiczne i analityczne, pozwala na rozwijanie kompetencji najbardziej pożądanych przez pracodawców, czyli:

- umiejętność rozwiązywania problemów krok po kroku za pomocą dzielenia problemów na mniejsze podproblemy,
- zdolność do uogólniania, testowania i oceny efektywności rozwiązania.

# PODSTAWOWE KONSTRUKCJE PROGRAMISTYCZNE

## SEKWENCJA POLECEŃ

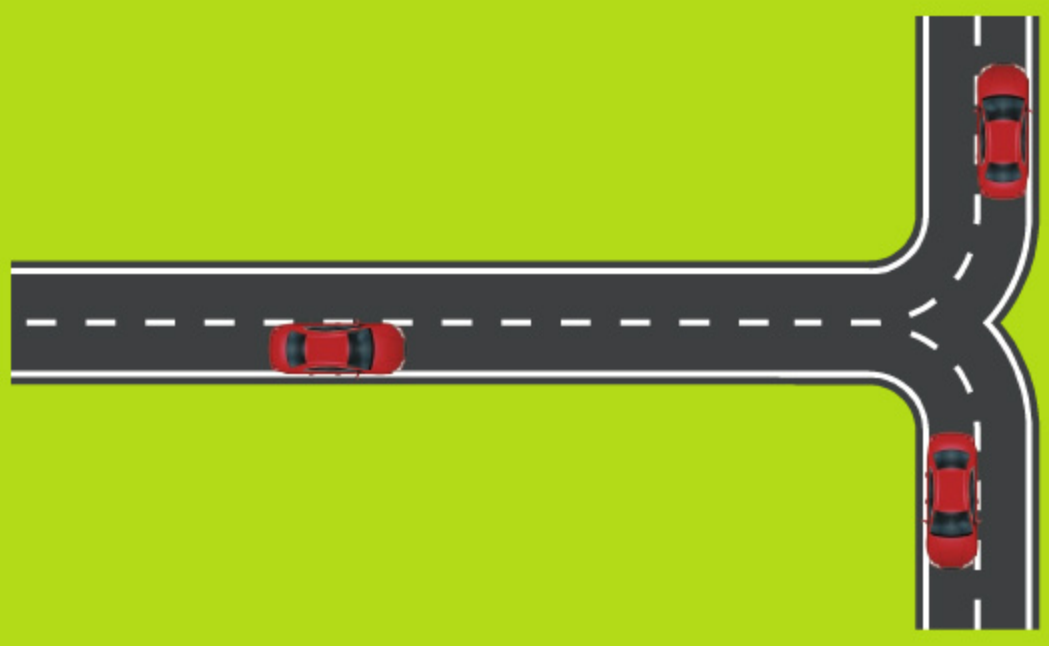
Polecenia są wykonywane jedno po drugim.



Polecenie dla kierowcy: jedź prosto.

## INSTRUKCJA WARUNKOWA

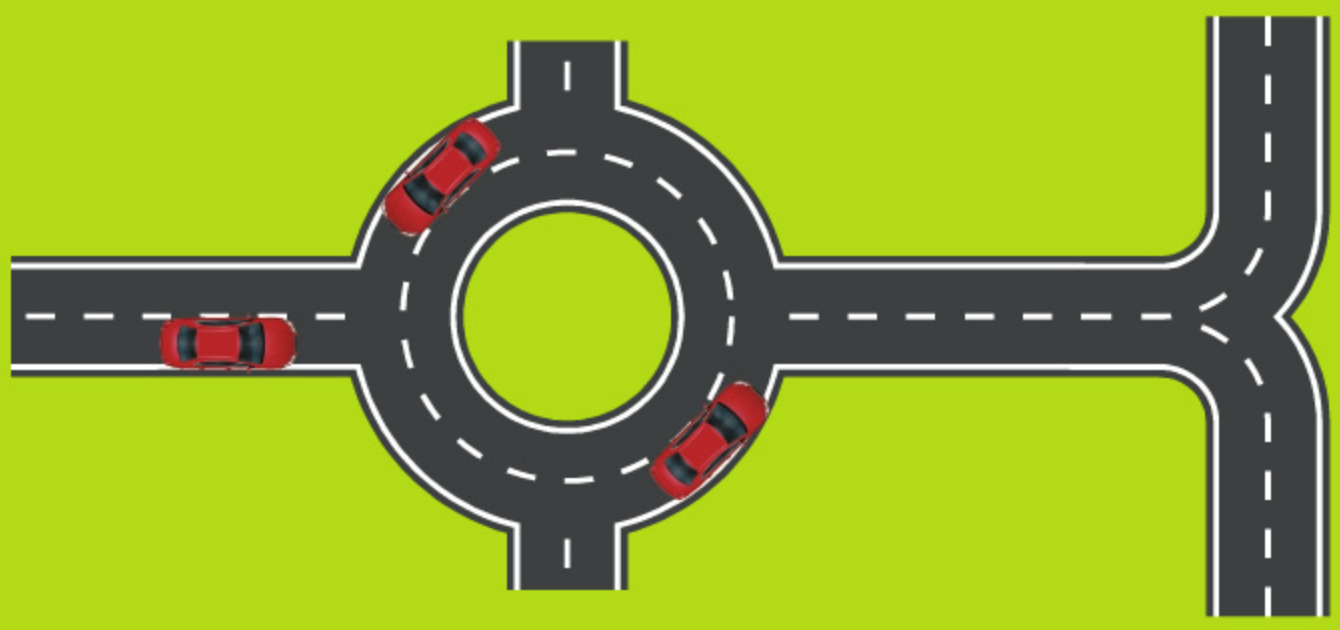
Pozwala na wykonanie różnych instrukcji w zależności od spełnienia określonego warunku.



Polecenie dla kierowcy: jeśli chcesz dojechać do szkoły, skręć w lewo; jeśli chcesz dotrzeć do sklepu, skręć w prawo.

## PĘTLA FOR

Powtarza zawarte w niej polecenia konkretną liczbę razy.



Polecenie dla kierowcy: na rondzie wykonaj trzy okrążenia.

## PĘTLA WHILE

Powtarza zawarte w niej polecenia, dopóki zachodzi podany warunek.

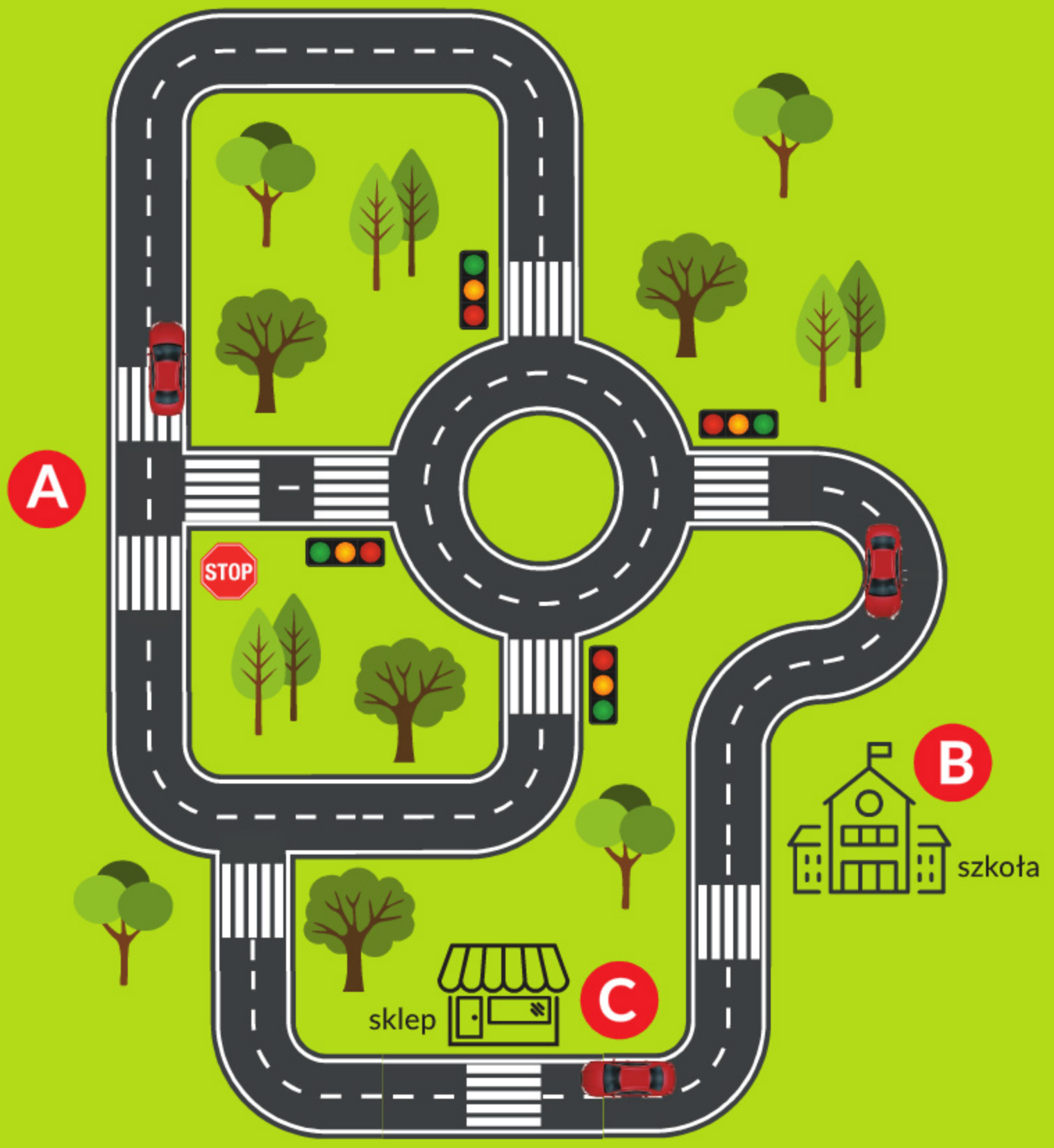


Polecenie dla kierowcy: na rondzie wykonuj okrążenia dopóty, dopóki zjazd nie zostanie odblokowany przez policjanta.

**Każdy język programowania ma swoją specyfikę, jednak pewne konstrukcje są wspólne dla wszystkich powszechnie używanych języków. Sprawdź, czy znasz podstawy, zanim zaczniesz uczyć się konkretnego języka.**

## FUNKCJA

Grupuje polecenia, które tworzą określoną logiczną całość i po wywołaniu funkcji są wykonywane w określonej kolejności.




Polecenie dla kierowcy: aby dojechać do sklepu (punkt C), jedź w kierunku szkoły (punkt B), a potem tak, jak prowadzi droga.



### PRACA Z INFOGRAFIKĄ

1. Które instrukcje powodują rozgałęzienie programu (różne ścieżki), a które nie?
2. Czym różnią się pętle `for` i `while`?



# 14. Podstawy pracy w środowisku Python

- Wprowadzenie do programowania w języku Python
- Wykonywanie obliczeń
- Definiowanie prostych funkcji

## ZADANIE NA START

**Programowanie w Pythonie**

Python to język interpretowany, którego używają zarówno początkujący programiści, jak i profesjonaliści. Doskonale sprawdza się jako narzędzie do tworzenia bardziej lub mniej złożonych aplikacji desktopowych, pracujących pod kontrolą systemów operacyjnych Windows, Linux czy OS, dynamicznych stron internetowych oraz zaawansowanych aplikacji webowych. Przykładowo: Google wykorzystuje Pythona w wyszukiwarce, amerykańska National Security Agency – w kryptografii oraz analizach wywiadowczych, firmy Intel, Hewlett-Packard czy IBM – w testowaniu urządzeń. Znajdź trzy aplikacje napisane w języku Python.




## ABC JĘZYKA PYTHON

W systemie Windows pracę w środowisku Python należy rozpocząć od zainstalowania tego środowiska na komputerze. W Ubuntu Linuksie ten język stanowi standardowy element systemu. Środowisko Python w wersji 3.* należy pobrać z serwisu **https://python.org**. Oprócz rekomendowanej opcji instalacji programu dla wszystkich użytkowników warto zaznaczyć opcję **Add Python 3.* to PATH**, umożliwiającą uruchamianie skryptów w wierszu polecenia z dowolnej lokalizacji.

**POSZUKAJ W INTERNECIE**

Poszukaj w internecie wywiadów z twórcą Pythona Guidem van Rossumem, aby dowiedzieć się więcej o tym języku i jego filozofii oraz o pochodzeniu samej nazwy.

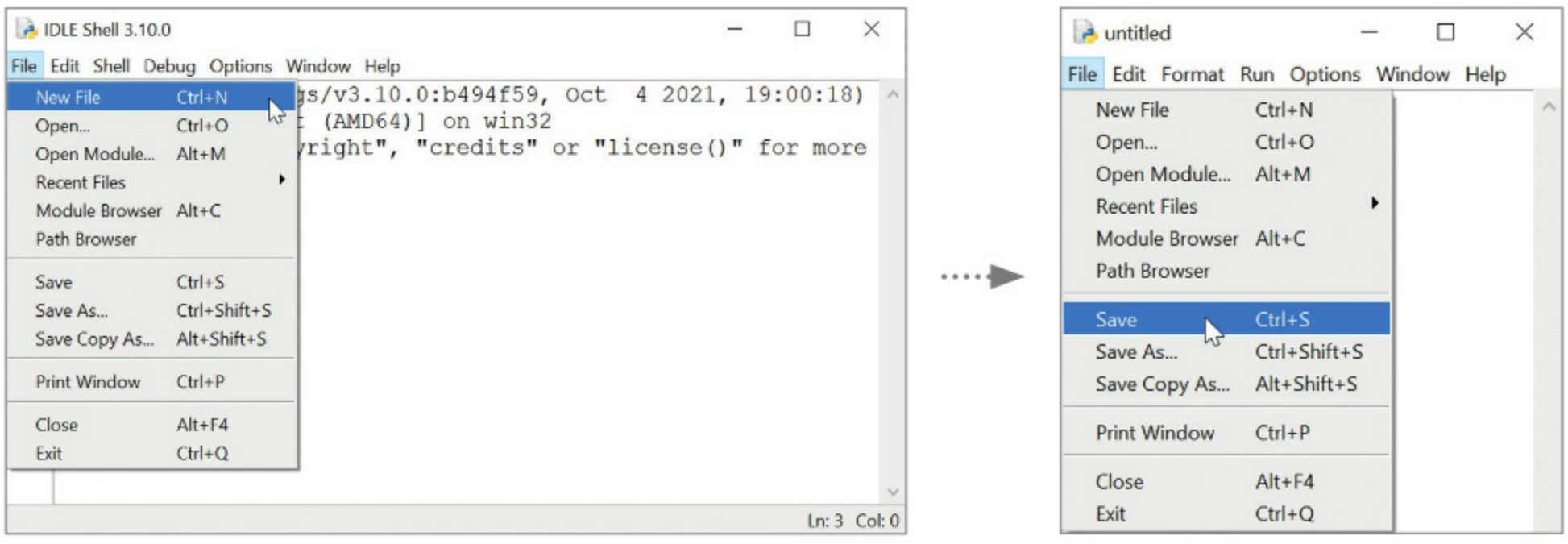
Częścią domyślnego pakietu instalacyjnego jest IDLE – zintegrowane środowisko deweloperskie przeznaczone do nauki dla osób początkujących.

Po uruchomieniu IDLE otworzy się okno Shell – powłoka systemowa działająca jako interpreter, który wypisuje wynik działania skryptu zapisanego w nim w trybie interaktywnym (tj. bezpośrednio w oknie Shell) albo w trybie skryptowym (tj. w edytorze kodu źródłowego). Trzy nawiasy ostrokątne, tzw. znak zachęty, oznaczają, że interpreter czeka na wprowadzenie polecenia.



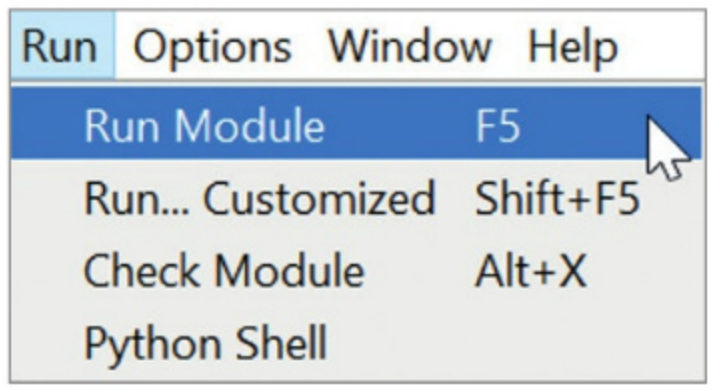

**Rys. 1.** Okno Shell edytora IDLE

Aby rozpocząć pracę w edytorze kodu źródłowego, należy kliknąć **File** na pasku menu i wybrać opcję **New File**. Otworzy się puste okno edytora o domyślnej nazwie **Untitled**. Następnie w oknie edytora należy ponownie kliknąć **File** i wskazać opcję **Save** lub **Save as**, nazwać plik i zapisać go w preferowanej lokalizacji. W ten sposób powstanie plik z rozszerzeniem .py.




**Rys. 2.** Tworzenie i zapisywanie pliku

Aby uruchomić program, należy wybrać na pasku menu **Run → Run Module**. Gdy pojawi się okno z pytaniem, czy zapisać zmiany, trzeba kliknąć **OK**. Warto pamiętać, że każde uruchomienie pliku Pythona skutkuje jego zapisaniem.




**Rys. 3.** Uruchamianie programu w IDLE

Aby ponownie edytować utworzony plik, trzeba go zaznaczyć, a następnie z menu podręcznego wybrać polecenie **Edit with IDLE** (plik nie otworzy się do edycji poprzez zwyczajowe dwukrotne kliknięcie myszą).

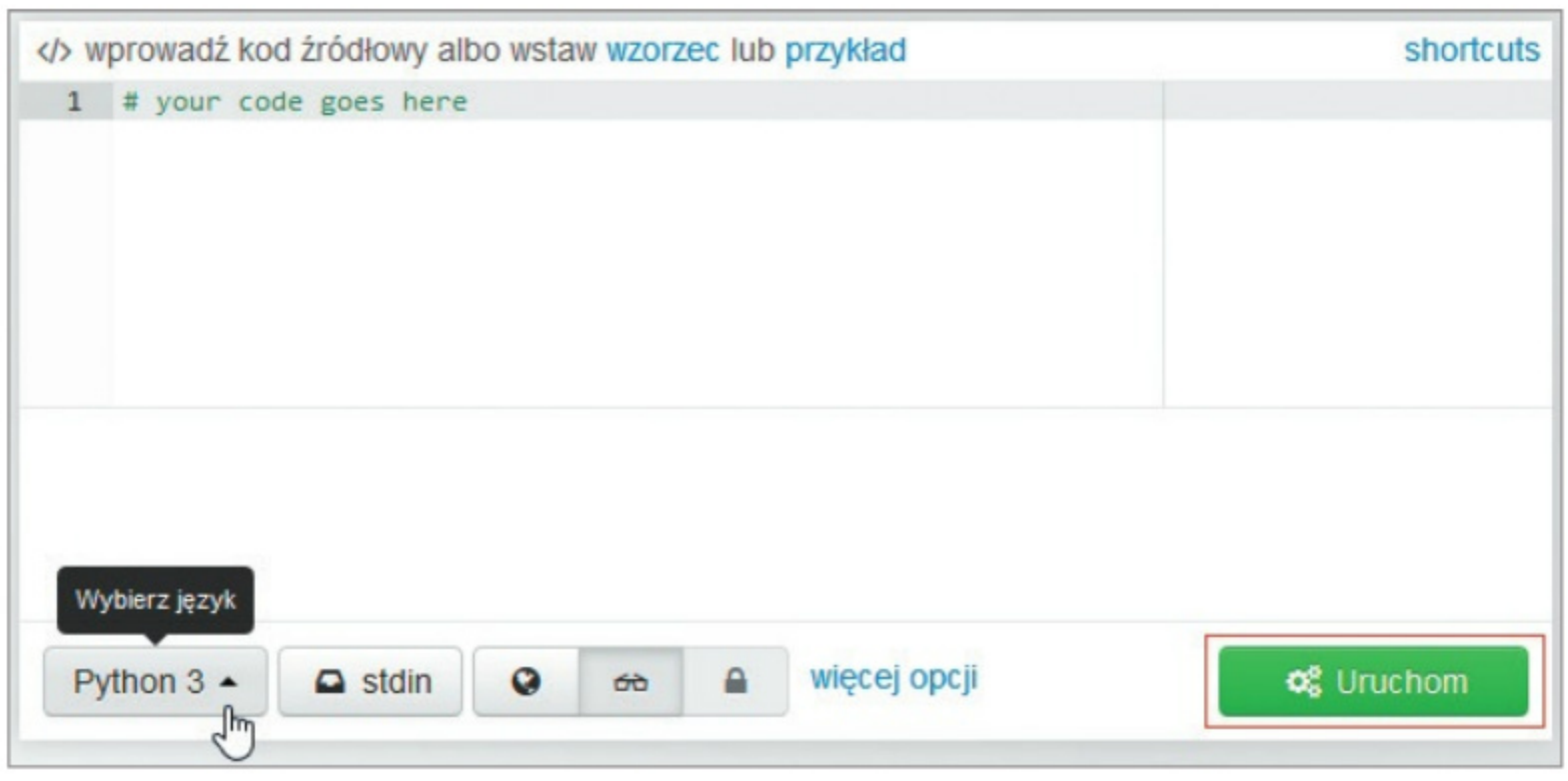



**Rys. 4.** Otwieranie pliku do edycji w IDLE

## ZAMIAST IDLE

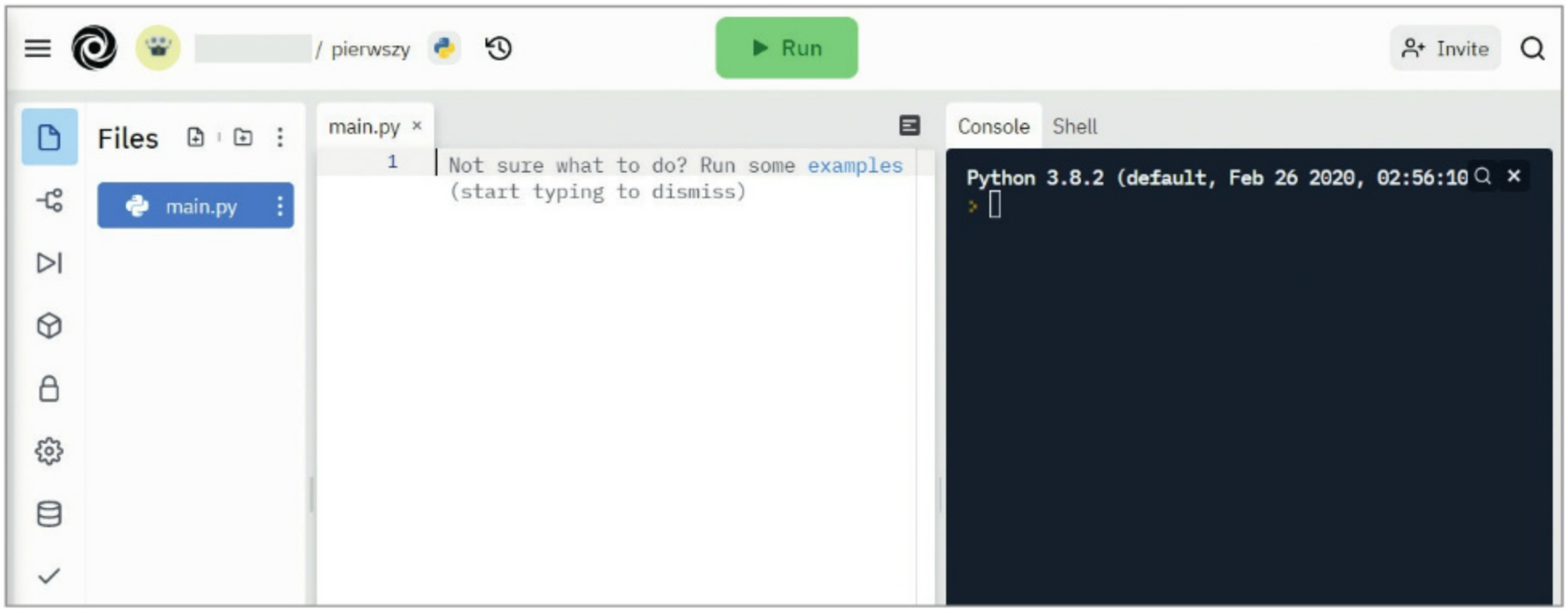
Skrypty Pythona można zapisywać w dowolnym edytorze tekstu, np. w zwykłym systemowym Notatniku, można też korzystać z coraz popularniejszych kompilatorów online, które obsługują wiele języków programowania (niektóre nawet ponad 50).

Pracę w chmurze umożliwia m.in. serwis **ideone.com** stworzony przez polską firmę z Trójmiasta. Wystarczy w przeglądarce wpisać adres serwisu, po wgraniu strony wybrać z rozwijanej listy odpowiedni język i można zacząć wpisywać kod. Aby uruchomić program, należy kliknąć przycisk **Uruchom**. Warto założyć konto i ustawić domyślny język programowania, język interfejsu oraz włączyć podświetlanie składni, a także używać notatek do opisu programu oraz etykiet do organizacji swoich kodów. Wszystkie skrypty programów są zapisywane w chmurze i znajdują się w sekcji **moje kody**. Można je pobrać na dysk i skorzystać z nich ponownie.




**Rys. 5.** Okno edytora w serwisie ideone.com: kursor wskazuje wybór języka, a czerwona ramka – uruchomienie programu

Osoby swobodnie posługujące się językiem angielskim mogą skorzystać z serwisu programistycznego **replit.com**. W tym przypadku konieczne jest założenie konta lub zalogowanie się poprzez konto Google. Aby rozpocząć pracę, należy kliknąć przycisk **+ Create Repl**, a następnie wybrać preferowany język, zatytułować projekt lub zaakceptować nazwę domyślną i zatwierdzić zmiany wprowadzone w oknie przyciskiem **+ Create Repl**. Kod wpisuje się w środkowym polu. Aby uruchomić program, należy kliknąć przycisk **Run**. Efekt działania pojawia się w oknie z prawej strony.




**Rys. 6.** Tworzenie nowego projektu w serwisie **replit.com**

## POLECENIE PRINT

Programowanie polega na pisaniu poleceń zgodnie z regułami obowiązującymi dla danego języka. Te polecenia w dalszej kolejności są analizowane przez interpreter, a następnie wykonywane przez komputer.

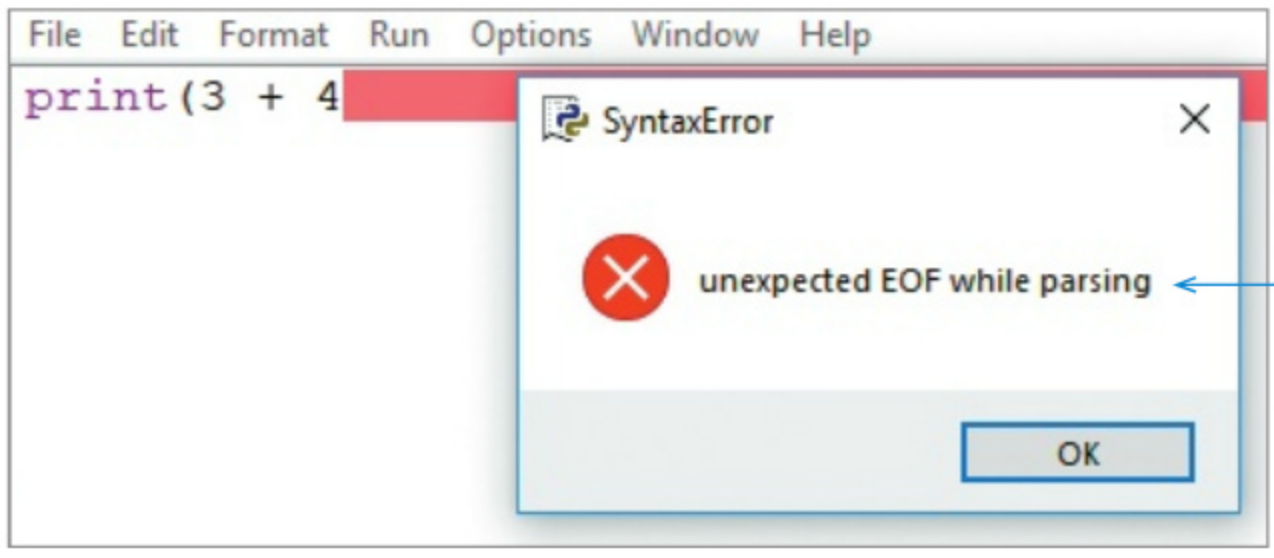
Funkcja **`print()`** służy do wypisywania na ekranie jej argumentów, czyli np. ciągu znaków, liczb, zmiennych albo wyników innej wywołanej funkcji.

**ZAPAMIĘTAJ!**

```
print("<ciąg znaków>")
print(<liczba>)
print(<zmienna>)
```

Aby dodać pusty wiersz, można skorzystać z polecenia **`print()`**.

Jeśli do zapisu wkradnie się błąd składni, np. zabraknie wcięcia lub nawiasu zamykającego, pojawi się komunikat **`SyntaxError`** – kolorowy pasek w IDLE wskaże, który wiersz należy skorygować. W przypadku błędów związanych z nazwami, np. dotyczącymi zmiennych czy funkcji, po uruchomieniu programu wyświetli się komunikat **`NameError`**.




**Rys. 7.** Przykładowe komunikaty błędów w IDLE i w serwisie **replit.com**

## Ćwiczenie 1. Wypisywanie tekstu

Wypisz na ekranie trzy wybrane przez siebie hasła na temat podróżowania. Każde z nich powinno wyświetlać się w nowym wierszu.

1. W edytorze kodu źródłowego wpisz polecenie **`print`** i otwórz nawias. W edytorach online nawias domknie się automatycznie, w IDLE musisz dodać samodzielnie nawias zamykający.

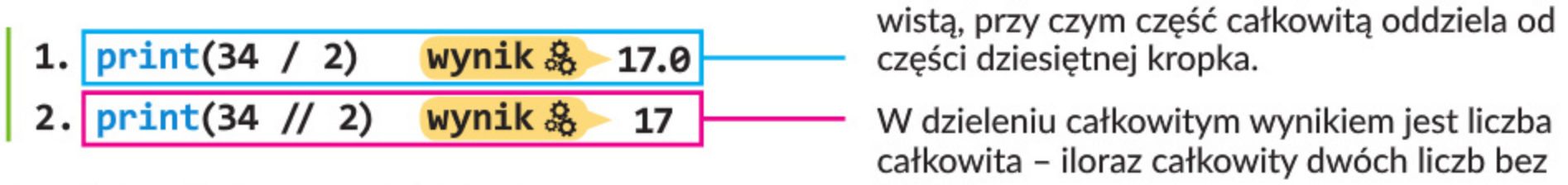
2. Wpisz w nawiasie hasło.

```
print("Zwiedzaj świat.")
print("Należy podróżować, by się czegoś nauczyć.")
print("Raz do roku pojedź w miejsce, w którym jeszcze nie byłeś.")
```

3. Uruchom program.

### OPERATORY

Python obsługuje operatory arytmetyczne i porównania. Do siedmiu podstawowych **operatorów arytmetycznych** należą: dodawanie (+), odejmowanie (–), mnożenie (*), potęgowanie (**), dzielenie (/), dzielenie całkowite (//) oraz reszta z dzielenia całkowitego (%).




**Rys. 8.** Przykłady operacji dzielenia

## Ćwiczenie 2. Podstawowe działania arytmetyczne

Wyznacz w jednym pliku kolejno: sumę, różnicę, iloczyn, iloraz, iloraz całkowity, resztę z dzielenia całkowitego oraz kwadrat liczby. Do obliczeń wykorzystaj liczby całkowite.

1. Dla każdego przykładu w edytorze kodu źródłowego wpisz polecenie **`print`** i dodaj nawiasy.

2. Wpisz w nawiasie odpowiedni wzór. Zwróć uwagę na kolejność wpisywania liczb.

   - Do wyznaczenia sumy wykorzystaj operator +.

     ```
     print(7 + 2)
     print(2 + 7)
     ```

   - Do wyznaczenia różnicy wykorzystaj operator –.

     ```
     print(7 - 2)
     print(2 - 7)
     ```

   - Do wyznaczenia iloczynu wykorzystaj operator *.

     ```
     print(7 * 2)
     print(2 * 7)
     ```

   - Do wyznaczenia ilorazu wykorzystaj operator /.

     ```
     print(4 / 2)
     print(7 / 3)
     ```

   - Do wyznaczenia ilorazu całkowitego wykorzystaj operator //.

     ```
     print(4 // 2)
     print(7 // 3)
     ```

   - Do wyznaczenia reszty z dzielenia całkowitego wykorzystaj operator %.

     ```
     print(19 % 2)
     print(14 % 3)
     print(15 % 3)
     ```

   - Do wyznaczenia kwadratu liczby wykorzystaj operator **.

     ```
     print(6 ** 2)
     ```

3. Wykonaj obliczenia w pamięci i uruchom skrypt.

**WIEM WIĘCEJ**

Środowiska Python można używać do wykonywania obliczeń na bardzo dużych liczbach. Takie działania nie są możliwe do zrealizowania na kalkulatorze prostym, którego wyświetlacz pozwala na pokazanie ośmiu cyfr, czyli np. do przeliczania lat świetlnych na kilometry: 1 rok świetlny to 9 460 730 777 119,56 km, a 15 lat świetlnych to 141 910 961 656 793,47 km.

## Ćwiczenie 3. Kupowanie obcej waluty

Oblicz, ile musisz zapłacić w złotych za 126 euro (aktualny kurs wymiany sprawdź na stronie NBP).

1. W edytorze kodu źródłowego wpisz polecenie **`print`** i dodaj nawiasy. W razie potrzeby domknij nawias samodzielnie.

2. Wpisz w nawiasie odpowiedni wzór – przemnóż daną kwotę przez aktualny kurs wymiany (pamiętaj, że w Pythonie separatorem części dziesiętnej i całkowitej jest kropka).

- Dla 1 euro wartego 4,60 zł kod przyjmuje postać:

```
1. print(126 * 4.6)
```

3. Uruchom program.

**Operatory porównania** wykorzystywane w języku Python zaprezentowano w poniższej tabeli wraz z przykładami. Wynikami porównań są wartości logiczne **`True`** (prawda) lub **`False`** (fałsz).

| Operatory porównania | | | |
|---|---|---|---|
| == | równe | `3 == 3 → True` | `4 == 3 → False` |
| != | różne | `3 != 2 → True` | `3 != 3 → False` |
| < | mniejsze | `3 < 4 → True` | `4 < 3 → False` |
| <= | mniejsze równe | `4 <= 4 → True` | `4 <= 3 → False` |
| > | większe | `4 > 3 → True` | `3 > 4 → False` |
| >= | większe równe | `4 >= 4 → True` | `4 >= 5 → False` |

## Ćwiczenie 4. Porównanie cen wyrażonych w różnych walutach

Te same buty w Grecji kosztują 49 euro, a w Polsce 200 zł. Sprawdź, czy buty kupione w Grecji są droższe niż buty kupione w Polsce (aktualny kurs wymiany znajdziesz na stronie NBP). Zastosuj operatory arytmetyczne i porównania.

1. W edytorze kodu źródłowego wpisz polecenie **`print`** i otwórz nawias. W razie potrzeby domknij nawias samodzielnie.

2. Wpisz w nawiasie odpowiedni wzór – przemnóż cenę butów w Grecji przez aktualny kurs wymiany (pamiętaj o zastąpieniu przecinka kropką) i porównaj z ceną butów w Polsce.

- Dla 1 euro wartego 4,60 zł kod może przyjąć postać:

```
print(49 * 4.6 >= 200)
```

3. Uruchom program.

- Dla 1 euro wartego 4,60 zł wynikiem jest **`True`**.

## ZMIENNE

Zmienne służą do przechowywania wartości określonego typu, którymi są m.in. liczby całkowite (typ **int**), liczby rzeczywiste (**float**), wartości logiczne (**bool**) i ciągi znaków (**string**). Każda zmienna w Pythonie powinna mieć krótką i znaczącą nazwę (bez polskich liter i spacji), np. **n** albo **liczba**. Należy przy tym pamiętać, że **wielkość liter jest istotna** – **N** i **n** to różne zmienne. Aby nadać zmiennej wartość, należy zastosować operator przypisania **=**.

```
1. x = 34
2. y = 66
3. print(x + y)
```

```
wynik
100
```

**Rys. 9.** Wypisanie sumy zmiennych całkowitych **x** oraz **y**

## Ćwiczenie 5. Średni dystans

W czasie wakacji uczniowie na obozie wędrownym przeszli w ciągu kolejnych trzech dni 12, 14 i 18 km. Zdefiniuj odpowiednie zmienne i oblicz średnią długość pokonywanych tras.

1. W edytorze kodu źródłowego zdefiniuj zmienne i przypisz im podane wartości.

```
1. a = 12
2. b = 14
3. c = 18
```

2. W kolejnym wierszu wpisz polecenie **print** i dodaj nawiasy.

3. Zastosuj odpowiedni wzór obliczania średniej, wypisz obliczaną wartość.

```
4. print((a + b + c) / 3)
```

4. Sprawdź odpowiedź – uruchom program.

## DEFINIOWANIE FUNKCJI

Funkcje pozwalają podzielić większy problem na mniejsze, względnie niezależne części, a także przyśpieszają i ułatwiają pisanie programu, który staje się dzięki nim bardziej zwięzły i przejrzysty. Definicję funkcji zaczyna się od słowa kluczowego **def**, następnie podaje się nazwę funkcji (bez polskich znaków, spacji) i wstawia nawiasy, w których można umieścić parametr lub parametry lub nie umieszczać niczego (funkcja bez parametru). Wiersz należy zakończyć dwukropkiem, który zagwarantuje, że po naciśnięciu klawisza **Enter** w następnym wierszu powstanie wcięcie. W kolejnych wierszach kodu podaje się instrukcje do wykonania. Polecenie **return** przekazuje szukaną wartość i kończy działanie funkcji.

**ZAPAMIĘTAJ!**

```
def <nazwa funkcji>(<parametr>):
    <instrukcje>
```

W Pythonie polecenia są grupowane w bloki, które uzyskuje się przez indentację, tj. wcięcia w wierszach. Jeśli kilka wierszy stanowi jeden blok, to wszystkie muszą być poprzedzone jednakowymi wcięciami.




```
1. def pole():
2.     a = 5
3.     return a ** 2
4.
5. print(pole())
```

**Rys. 10.** Funkcja **pole()** i jej wywołanie




```
1. def pole(a):
2.     return a ** 2
3.
4. print(pole(5))
```

**Rys. 11.** Funkcja **pole(a)** i jej wywołanie

## Ćwiczenie 6. Przeliczanie prędkości

Samolot z Warszawy do Rzymu pokonuje dystans 1320 km. Przeanalizuj dane w tabeli i zdefiniuj funkcję **jak_szybko(t)**, której parametrem jest czas przelotu samolotu w godzinach, a wynikiem – średnia prędkość tego samolotu.

| Wywołanie funkcji | Wynik |
|---|---|
| **jak_szybko(2.5)** | **528.0** |
| **jak_szybko(3)** | **440.0** |

1. Przeanalizuj podane przykłady.

- Aby wyznaczyć średnią prędkość, należy podzielić drogę przez czas.

2. Zdefiniuj funkcję **jak_szybko(t)**, np.

```
1. def jak_szybko(t):
2.     return 1320 / t
```

```
1. def jak_szybko(t):
2.     s = 1320
3.     return s / t
```

3. Sprawdź wyniki funkcji dla różnych danych.

```
3. print(jak_szybko(2.5))
4. print(jak_szybko(3))
5. print(jak_szybko(2.2))
```

4. Uruchom program.

## Ćwiczenie 7. Przeliczanie jednostek

Mila lądowa, jednostka długości stosowana w krajach anglosaskich, jest równa 1,609 km. Przeanalizuj dane w tabeli i zdefiniuj funkcję **zamiana(m)**, której parametrem jest odległość w milach lądowych, a wynikiem – ta sama odległość w kilometrach.

| Wywołanie funkcji | Wynik |
|---|---|
| **zamiana(10)** | **16.09** |
| **zamiana(12)** | **19.308** |

1. Przeanalizuj podane przykłady.

- Aby zamienić mile na kilometry, należy pomnożyć odległość w milach przez 1,609.

2. Zdefiniuj funkcję **zamiana(m)**, np.

```
1. def zamiana(m):
2.     return m * 1.609
```

3. Sprawdź wyniki funkcji dla różnych danych.

```
3. print(zamiana(10))
4. print(zamiana(12))
5. print(zamiana(1))
```

4. Uruchom program.

## Ćwiczenie 8. O której godzinie pociąg dotrze do celu

Nocny pociąg z Krakowa do Paryża wyjeżdża ze stacji o godzinie 21.00. Przeanalizuj dane w tabeli i zdefiniuj funkcję **godzina(n)**, której parametrem jest czas przejazdu pociągu w godzinach, a wynikiem – godzina dojazdu do celu.

| Wywołanie funkcji | Wynik |
|---|---|
| **godzina(14)** | **11** |
| **godzina(20)** | **17** |

1. Przeanalizuj podane przykłady i zapisz słownie odpowiednią formułę.

- 21 + 14 = 35, 21 + 20 = 41. Doba trwa 24 godziny, dlatego aby obliczyć czas przyjazdu pociągu, należy wyznaczyć resztę z dzielenia wyniku dodawania przez 24.

2. W edytorze kodu źródłowego zdefiniuj funkcję **godzina(n)** – wykorzystaj do obliczeń resztę z dzielenia całkowitego.

```
1. def godzina(n):
2.     return (21 + n) % 24
```

3. Sprawdź działanie skryptu z różnymi danymi.

**WIEM WIĘCEJ**

Podczas pisania programu należy dbać o odpowiednie **standardy kodowania**. Z regułami i zaleceniami dotyczącymi formatowania kodu w Pythonie (*Style Guide for Python Code*) można się zapoznać na stronie **www.python.org/dev/peps/pep-0008**.

## Ćwiczenie 9. Która godzina będzie w Nowym Jorku

Sprawdź, jaka strefa czasowa obowiązuje w Nowym Jorku, a następnie przeanalizuj dane w tabeli i zdefiniuj funkcję **`godz_nj(g)`**, której parametrem jest pełna godzina w Warszawie, a wynikiem – aktualna pełna godzina w Nowym Jorku.

| Wywołanie funkcji | Wynik |
|---|---|
| **`godz_nj(18)`** | **`12`** |
| **`godz_nj(2)`** | **`20`** |

## PODSUMOWANIE

- Python to język interpretowany o przejrzystej składni, za pomocą którego można pisać mniej i bardziej skomplikowane programy. Jego skrypty są analizowane przez interpreter, a następnie wykonywane przez komputer.
- Funkcja **`print()`** służy do wypisywania na ekranie jej argumentów, którymi mogą być napisy, liczby, zmienne oraz wyniki innej wywołanej funkcji.
- Podstawowe operatory arytmetyczne i porównania są zbliżone do używanych w języku naturalnym, dlatego korzystanie z nich szybko staje się intuicyjne.
- Zmienne służą do przechowywania wartości określonego typu, takich jak liczby całkowite i rzeczywiste, wartości logiczne, napisy. Należy im nadawać krótkie i znaczące nazwy.
- Polecenia są grupowane w bloki, które uzyskuje się przez wcięcia w wierszach (indentację).
- Język Python rozróżnia małe i wielkie litery.
- Rozwiązywany problem warto podzielić na mniejsze problemy, które można niezależnie rozwiązać przez definiowanie funkcji.
- Definicję funkcji zaczyna się od słowa kluczowego **`def`**, następnie podaje się nazwę funkcji i wstawia nawiasy, w których można umieścić parametr. Deklarację należy zakończyć dwukropkiem. W kolejnych wierszach kodu podaje się instrukcje do wykonania. Polecenie **`return`** przekazuje szukaną wartość i kończy działanie funkcji.

## PYTANIA SPRAWDZAJĄCE

1. Jaka liczba będzie wynikiem operacji **`5 / 2`**, a jaka – wynikiem operacji **`5 // 2`**?
2. Czym się różni polecenie **`print("a")`** od polecenia **`print(a)`**?
3. Do czego służy polecenie **`return`** w funkcji?

## PYTANIA QUIZOWE

1. Co jest wynikiem działania **`121 % 3`**?
   - **a.** 12
   - **b.** 1
   - **c.** 21
   - **d.** 40
2. Co wypisze polecenie **`print()`**?
   - **a.** Spację.
   - **b.** Dwie spacje.
   - **c.** Cudzysłów.
   - **d.** Pusty wiersz.
3. W jaki sposób działa polecenie **`return`**?
   - **a.** Wypisuje szukaną wartość.
   - **b.** Oblicza wynik funkcji.
   - **c.** Przekazuje szukaną wartość – wynik funkcji.
   - **d.** Zamienia liczbę na napis.

## ZADANIA DODATKOWE

### Zadanie 1. Która godzina będzie w Warszawie

Różnica czasu między Los Angeles a Warszawą wynosi dziewięć godzin. Gdy w Los Angeles zegar wskazuje godzinę 21.00, to w Warszawie jest godzina 6 rano. Przeanalizuj dane w tabeli i zdefiniuj funkcję **`godz_waw(g)`**, której parametrem jest pełna godzina w Los Angeles, a wynikiem – aktualna godzina w Warszawie.

| Wywołanie funkcji | Wynik |
|---|---|
| **`godz_waw(8)`** | **17** |
| **`godz_waw(21)`** | **6** |

### Zadanie 2. Ile to kosztuje

Zdefiniuj funkcję **`cena(f)`**, której parametrem jest cena w dolarach funta danego produktu, a wynikiem – cena w złotych kilograma tego produktu. Sprawdź w internecie aktualny kurs dolara oraz przelicznik jednostek masy (funt i kilogram).



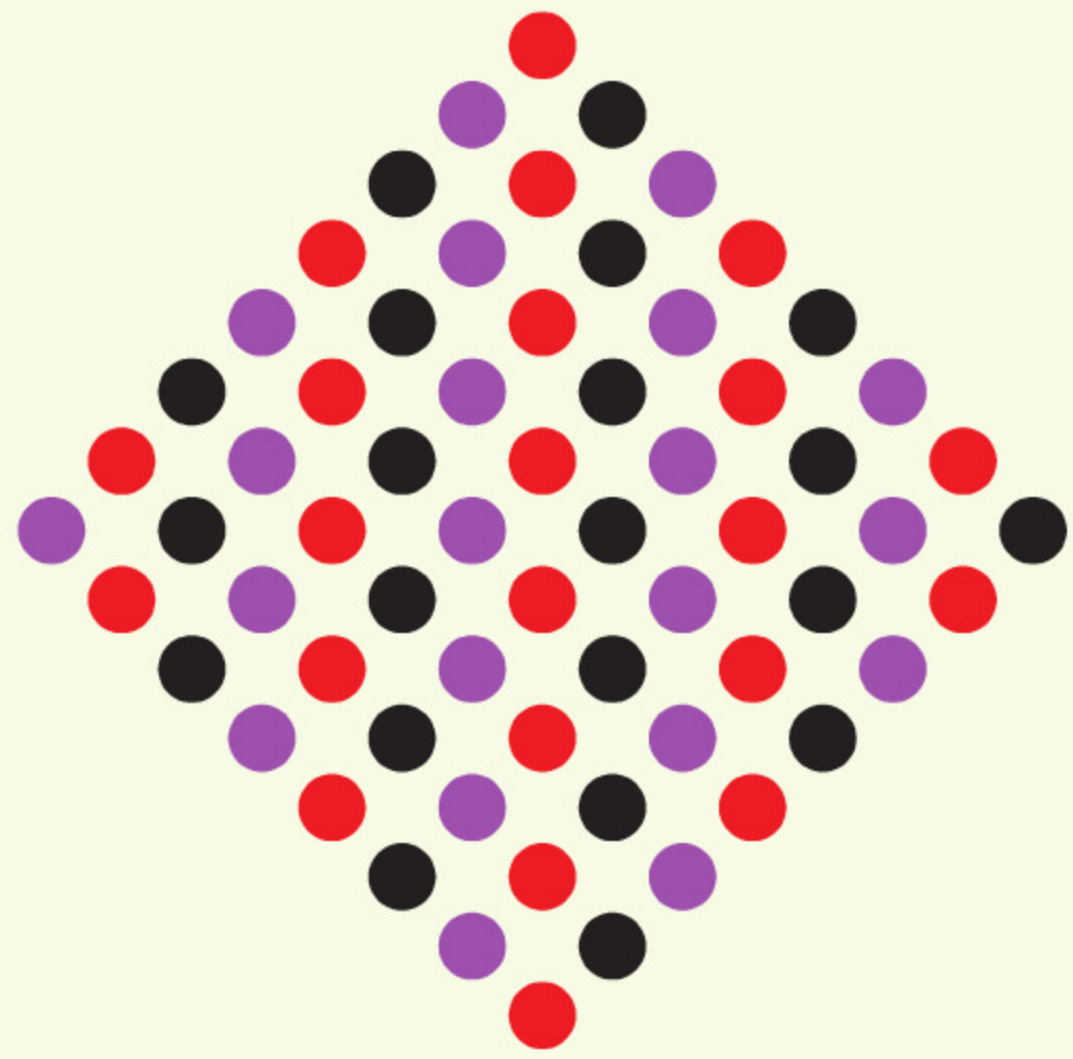
# 15. Definiowanie funkcji obliczeniowych

- Stosowanie instrukcji warunkowych
- Wykonywanie iteracji za pomocą pętli **for**
- Analizowanie i testowanie rozwiązań

## ZADANIE NA START

### Wzór z korali

Z korali w trzech kolorach można układać ciekawe wzory. Przyjrzyj się poniższemu rysunkowi i określ regułę dotyczącą kolorów korali. Policz, ile jest korali każdego z kolorów, a następnie napisz wzór na ich liczbę w zależności od liczby korali na boku kwadratu.



## ROZWIĄZYWANIE PROBLEMÓW W PYTHONIE

Podczas rozwiązywania problemów programistycznych należy najpierw postarać się zrozumieć problem, następnie przeanalizować go na kilku przykładach, potem zaproponować sposób wykonania obliczeń i zapisać go zgodnie z regułami obowiązującymi w danym środowisku, wreszcie przetestować.

Zapis formalny zwykle wymaga zdefiniowania funkcji zawierających mniej lub bardziej złożone sekwencje poleceń. W tabeli zamieszczonej na następnej stronie przypomniano konstrukcje stosowane w języku Python, omówione podczas poprzedniej lekcji, oraz zaprezentowano składnię dwóch kolejnych przydatnych konstrukcji – instrukcji warunkowej i pętli **for**.

| Konstrukcja | Składnia | Przykład |
|---|---|---|
| instrukcja przypisania | `nazwa_zmiennej = wartość` | `x = 3` |
| funkcja **`print()`** | `print(argument)` | `print("tekst")`<br>`print(3)`<br>`print(x)`<br>`print(pole_kwadratu(12))` |
| definiowana funkcja | `def nazwa(parametry):`<br>`    instrukcje` | `def pole_prostokata(bok1, bok2):`<br>`    return bok1 * bok2` |
| instrukcja warunkowa prosta | `if warunek:`<br>`    instrukcje` | `if x % 3 == 0:`<br>`    print("liczba podzielna przez 3")` |
| instrukcja warunkowa złożona | `if warunek:`<br>`    instrukcje`<br>`elif warunek:`<br>`    instrukcje`<br>`else:`<br>`    instrukcje` | `if x > 0:`<br>`    print("liczba dodatnia")`<br>`elif x == 0:`<br>`    print("zero")`<br>`else:`<br>`    print("liczba ujemna")` |
| pętla **`for`** | `for nazwa_zmiennej in sekwencja:`<br>`    instrukcje` | `for i in range(10):`<br>`    print(i)` |

## Ćwiczenie 1. Ile korali

Przeanalizuj dane w tabeli i zdefiniuj funkcję **`ile(k)`**, której parametrem jest liczba korali na boku kwadratu **k**, a wynikiem – łączna liczba korali we wzorze.

| Wywołanie funkcji | Wynik |
|---|---|
| **`ile(5)`** | **25** |
| **`ile(6)`** | **36** |

1. Przeanalizuj treść ćwiczenia i podane przykłady. Wykonaj rysunki pomocnicze.



Jeśli bok kwadratu jest zbudowany z 5 korali, to łączna liczba korali tworzących kwadrat wynosi $5 \cdot 5 = 25$.



Jeśli bok kwadratu jest zbudowany z 6 korali, to łączna liczba korali tworzących kwadrat wynosi $6 \cdot 6 = 36$.

2. Zapisz odpowiednią formułę.

```
k * k
```

3. Zdefiniuj funkcję **ile(k)**.

```
1. def ile(k):
2.     return k * k
```

4. Sprawdź działanie skryptu z różnymi danymi.

## INSTRUKCJA WARUNKOWA

Instrukcja warunkowa pozwala na wykonanie różnych obliczeń w zależności od tego, czy zdefiniowane wyrażenie logiczne jest prawdziwe, czy fałszywe.

- Jeśli liczba jest parzysta, podziel wartość przez 2, w przeciwnym razie odejmij 1.
- Jeśli liczba jest dodatnia, przypisz zmiennej wartość 1, gdy ujemna, przypisz –1, w przeciwnym razie przypisz 0.

Instrukcja warunkowa może składać się z jednej, dwóch lub trzech części.

- Na początku zawsze znajdują się słowo kluczowe **if** oraz warunek lub wyrażenie, które mogą mieć wartość **True** (1 lub inna wartość niezerowa) lub **False** (0). Wiersz należy zakończyć dwukropkiem.
- Potem następuje instrukcja albo blok instrukcji do wykonania w przypadku spełnienia warunku.
- Słowo kluczowe **elif** (skrót od *else if*) wprowadza nowy warunek lub wyrażenie oraz instrukcję do wykonania w przypadku spełnienia tego warunku.
- Słowo kluczowe **else** wprowadza instrukcję do wykonania, gdy żadne z powyższych warunków nie są prawdziwe.

**ZAPAMIĘTAJ!**

```
if <wyrażenie jest prawdziwe>:
     <wykonaj instrukcję 1>
elif <poprzednie wyrażenie jest fałszywe, ale to jest prawdziwe>:
     <wykonaj instrukcję 2>
else:
     <wykonaj instrukcję 3>
```

Instrukcja warunkowa na rys. 1 składa się z dwóch części. Można ją odczytać następująco:

- jeśli reszta z dzielenia całkowitego zmiennej **x** przez 2 jest równa 0, przypisz zmiennej **x** połowę przechowywanej przez nią liczby;
- w przeciwnym razie przypisz zmiennej **x** liczbę o 1 mniejszą.

```
1. if x % 2 == 0:
2.     x = x / 2
3. else:
4.     x = x - 1
```

**Rys. 1.** Przykładowa instrukcja warunkowa złożona z dwóch elementów

Instrukcja warunkowa na rys. 2 składa się z trzech części. Można ją odczytać następująco:

- jeśli zmienna **x** jest większa od 0, przypisz zmiennej **zn** wartość 1;
- jeśli poprzedni warunek jest fałszywy i zmienna **x** jest mniejsza od 0, przypisz zmiennej **zn** wartość –1;
- w przeciwnym razie przypisz zmiennej **zn** wartość 0.

```
1. if x > 0:
2.     zn = 1
3. elif x < 0:
4.     zn = -1
5. else:
6.     zn = 0
```

**Rys. 2.** Przykładowa instrukcja warunkowa złożona z trzech elementów

## Ćwiczenie 2. Dwukolorowy romb

Przeanalizuj dane w tabeli i zdefiniuj funkcję **`ile_kolor(k, kolor)`**, której parametrami są liczba korali na boku kwadratu **k** i kolor korali – **c** (czerwony) lub **n** (niebieski), a wynikiem jest liczba korali podanego koloru we wzorze.

| Wywołanie funkcji | Wynik |
|---|---|
| `ile_kolor(6, "c")` | **18** |
| `ile_kolor(7, "c")` | **24** |
| `ile_kolor(6, "n")` | **18** |
| `ile_kolor(7, "n")` | **25** |

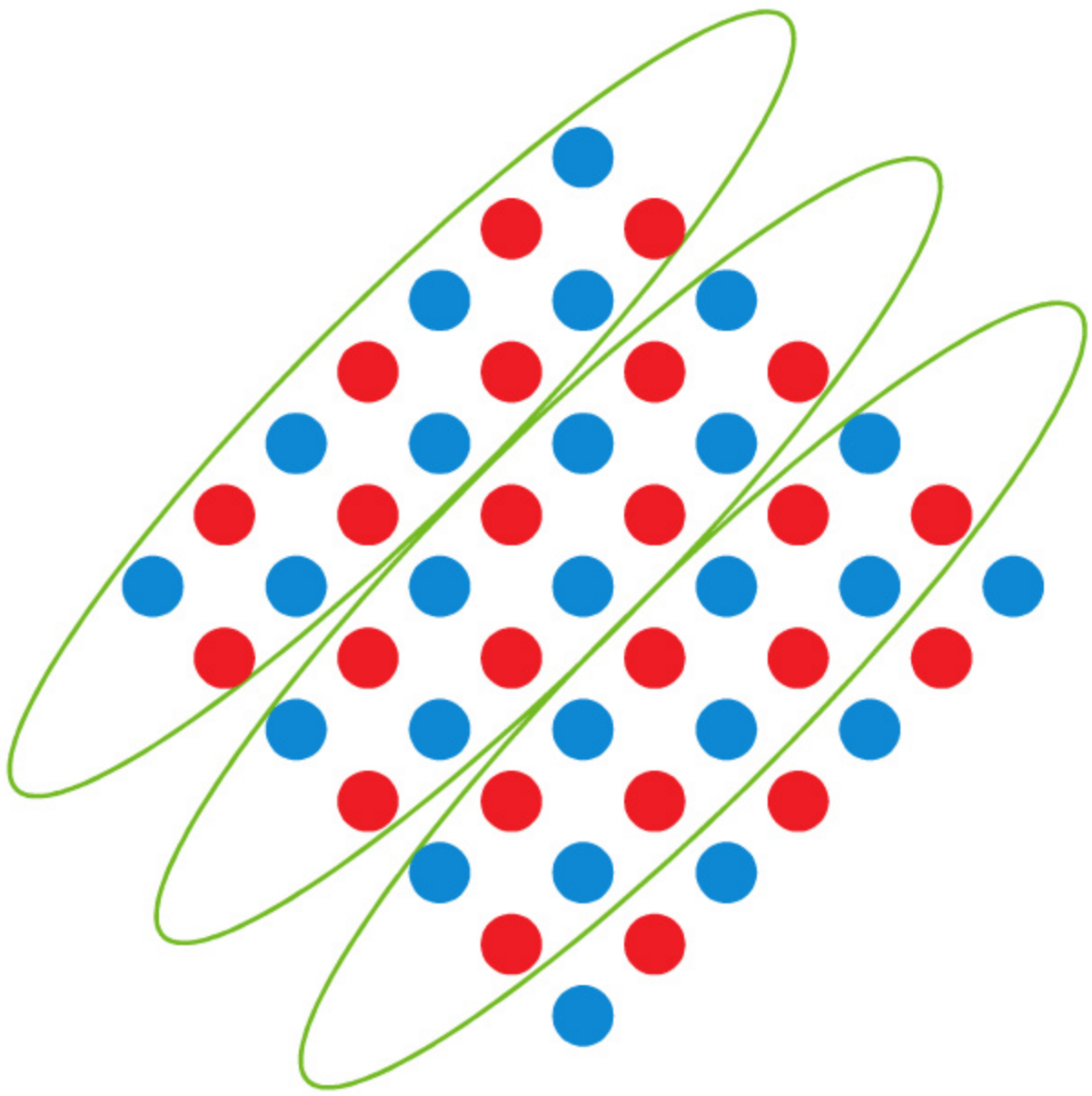
1. Przeanalizuj treść ćwiczenia i podane przykłady. Możesz wykonać rysunki pomocnicze.

- Dla wartości parzystych parametru **k** liczba korali dla obu kolorów jest jednakowa. Aby wyznaczyć liczbę korali w danym kolorze, trzeba pomnożyć przez siebie liczbę korali tworzących bok kwadratu i podzielić przez 2: **`6 * 6 // 2 = 18`**.



*k* = 6

- Dla wartości nieparzystych parametru **k** liczba korali czerwonych jest o 1 mniejsza od liczby korali niebieskich. Aby wyznaczyć liczbę korali w danym kolorze, trzeba wprowadzić dwa warunki:
  - jeśli korale są czerwone, wynikiem będzie liczba korali tworzących bok kwadratu, pomnożona przez siebie i podzielona przez 2: **7 * 7 // 2**;
  - w przeciwnym razie (tj. dla korali niebieskich) wynikiem będzie liczba korali tworzących bok kwadratu, pomnożona przez siebie i pomniejszona o 1, a następnie podzielona przez 2 i powiększona o 1.



*k* = 7

Połącz w pary dwa sąsiednie wiersze z wyjątkiem ostatniego, a następnie wyznacz osobno liczbę korali czerwonych i niebieskich w każdej parze wierszy i w wierszu ostatnim.

Korale czerwone:
(3 + 4) + (3 + 4) + (3 + 4) + 3 = 24.
Zapis, którego podstawę stanowi liczba 7:
24 = 7 * 7 // 2.
Zapis uogólniony: k * k // 2.

Korale niebieskie:
(4 + 3) + (4 + 3) + (4 + 3) + 4 = 25.
Zapis, którego podstawę stanowi liczba 7:
25 = (7 * 7 - 1) // 2 + 1.
Zapis uogólniony: (k * k – 1) // 2 + 1.

2. Zapisz odpowiednie formuły. Zauważ, że wykorzystują wzór zastosowany w ćwiczeniu 1.

- Liczba korali czerwonych:

```
k * k // 2
```

- Liczba korali niebieskich:

```
(k * k - 1) // 2 + 1
```

3. Zdefiniuj funkcję **ile_kolor(k, kolor)**.

```
1. def ile_kolor(k, kolor):
2.     if kolor == "c":
3.         return k * k // 2
4.     else:
5.         return (k * k - 1) // 2 + 1
```

4. Sprawdź działanie skryptu z różnymi danymi, w tym z parzystym i nieparzystym rozmiarem wzoru.

## PĘTLA FOR

Instrukcja iteracji służy do powtarzania jakiejś instrukcji lub bloku instrukcji w programie, np. dodawania wielu liczb do siebie. Dla każdego elementu sekwencji jest wykonywany podany ciąg instrukcji.

**ZAPAMIĘTAJ!**

```
for <nazwa zmiennej> in <sekwencja>:
    <instrukcje>
```

W przypadku pętli **for** używa się zmiennej sterującej pętlą, zwykle o nazwie **i**, a jako sekwencji funkcji **range(od, do, krok)**. Funkcja **range()** może mieć jeden parametr „do" (koniec zakresu), dwa parametry „od" i „do" (początek i koniec zakresu) albo trzy parametry „od", „do" i „krok" (początek zakresu, koniec zakresu i krok, o jaki zwiększa się kolejne wartości), np.

- **range(7)** – definiuje zakres od 0 do 6;
- **range(3, 7)** – definiuje zakres od 3 do 6, gdyż ostatnia generowana wartość to największa liczba całkowita mniejsza niż ograniczenie górne;
- **range(1, 7, 2)** – definiuje liczby od 1, nie większe niż 6 i co 2, a więc: 1, 3 i 5.

```
1. for i in range(3):
2.   print(i)
```

wynik

```
0
1
2
```

**Rys. 3.** Pętla wypisująca liczby od 0 do 2

## Ćwiczenie 3. Piramida

Przeanalizuj dane w tabeli i zdefiniuj funkcję **`piramida(k)`**, której parametrem jest liczba korali w podstawie **`k`**, a wynikiem – łączna liczba korali.

| Wywołanie funkcji | Wynik |
|---|---|
| **`piramida(5)`** | **15** |
| **`piramida(6)`** | **21** |

Zauważ, że ćwiczenie można wykonać na dwa sposoby.

### Sposób 1

1. Przeanalizuj treść ćwiczenia i podane przykłady.

- Rozpatrz liczbę korali i zapamiętaj ją w zmiennej o nazwie **`licznik`**. Najpierw **`licznik`** jest równy **`0`**. Potem zwiększa się o **`1`**, czyli liczbę korali w najwyższym rzędzie. W dalszej kolejności zwiększa się o **`2`**, tj. liczbę korali w rzędzie drugim od góry. Potem o **`3`**...



1 + 2 + 3 + 4 + 5 = 15



1 + 2 + 3 + 4 + 5 + 6 = 21

2. Zapisz odpowiednie formuły.

```
licznik = 0
licznik = licznik + 1
licznik = licznik + 2
...
licznik = licznik + k
```

3. Przeanalizuj powyższy zapis i uogólnij go – zastosuj pętlę.

```
licznik = 0
for i in range(1, k + 1):
    licznik = licznik + i
```

4. Zdefiniuj funkcję **`piramida(k)`**.

```
1. def piramida(k):
2.     licznik = 0
3.     for i in range(1, k + 1):
4.         licznik = licznik + i
5.     return licznik
```

5. Sprawdź działanie skryptu z różnymi danymi.

### Sposób 2

1. Przeanalizuj treść ćwiczenia i podane przykłady.

- Zauważ, że rzędy można pogrupować parami – pierwszy z ostatnim, drugi z przedostatnim itd.



$k = 5$

$1 + 2 + 3 + 4 + 5 = 15$

Liczby w parach sumują się do $k + 1$, co oznacza, że średnio rząd ma $(k + 1) / 2$ korali.



$k = 6$

$1 + 2 + 3 + 4 + 5 + 6 = 21$

2. Zapisz ogólną zależność wzorem. Ponieważ $k$ albo $k + 1$ jest zawsze liczbą parzystą, zastosuj dzielenie całkowite.

```
k * (k + 1) // 2
```

3. Zdefiniuj funkcję **piramida(k)**.

```
1. def piramida(k):
2.     return k * (k + 1) // 2
```

4. Sprawdź działanie skryptu z różnymi danymi.

## Ćwiczenie 4. Piramida – porównanie funkcji

Porównaj działanie obu funkcji z ćwiczenia 3 dla różnych parametrów.

1. Zapisz funkcje z poprzedniego ćwiczenia w jednym pliku pod nazwami **piramida1(k)** oraz **piramida2(k)**, a następnie skonstruuj polecenie, które wypisze na ekranie wyniki dla obu funkcji.

- Kod wypisujący wyniki dla parametrów w zakresie od 1 do 19 może przyjąć postać:

```
1. for i in range(1, 20):
2.     print(piramida1(i), piramida2(i))
```

2. Podsumuj swoje spostrzeżenia. Zastanów się, który wariant analizy problemu był dla ciebie bardziej czytelny (może jeden z nich sam się narzucał?). Czy przejście od rozpoznania prawidłowości do uogólnienia nie sprawiło ci trudności? Jakie dostrzegasz zalety i wady każdej z przyjętych strategii?

### Ćwiczenie 5. Trzy kolory

Zaprojektuj własny wzór złożony z koralików w co najmniej trzech kolorach. Napisz funkcję, której wynikiem będzie liczba korali każdego koloru.

## PODSUMOWANIE

- Instrukcja warunkowa pozwala na wykonanie różnych obliczeń w zależności od tego, czy zdefiniowane wyrażenie logiczne jest prawdziwe (ma wartość **`True`**, 1 lub inną wartość niezerową), czy fałszywe (ma wartość **`False`** lub 0).
- Instrukcja warunkowa może składać się z jednej lub kilku części. Na początku znajdują się słowo kluczowe **`if`** oraz warunek. Wiersz należy zakończyć dwukropkiem. Potem następuje instrukcja lub blok instrukcji do wykonania w razie spełnienia warunku. Słowo kluczowe **`elif`** wprowadza nowy warunek lub wyrażenie oraz instrukcję do wykonania w razie spełnienia tego warunku. Słowo kluczowe **`else`** wprowadza instrukcję do wykonania, gdy żaden z powyższych warunków nie jest prawdziwy.
- Instrukcja iteracji służy do powtarzania jakiejś instrukcji lub bloku instrukcji. Dla każdego elementu sekwencji wykonuje się podany zestaw instrukcji.
- W przypadku pętli **`for`** używa się zmiennej sterującej pętlą zwykle o nazwie **`i`**, a jako sekwencji – funkcji **`range(od, do, krok)`**.
- Funkcja **`range()`** może mieć jeden parametr „do” (koniec zakresu), dwa parametry „od” i „do” (początek i koniec zakresu) albo trzy parametry „od”, „do” i „krok” (początek zakresu, koniec zakresu i krok, o jaki zwiększa się kolejne wartości).

## PYTANIA SPRAWDZAJĄCE

1. Czy instrukcja warunkowa musi zawierać słowo kluczowe **`else`**? Uzasadnij.
2. Jak w języku Python zapisać instrukcję iteracji, aby zmienna sterująca pętlą zmieniała wartości od 10 do 0 włącznie?
3. Z jakimi wartościami parametrów warto testować każde zadanie?

## PYTANIA QUIZOWE

1. Jaka ostatnia wartość zostanie wypisana po wykonaniu poniższej pętli?

```
1. for i in range(3, 11, 2):
2.     print(i)
```

   **a.** 8
   **b.** 9
   **c.** 10
   **d.** 11

2. Dla jakich liczb wynikiem poniższej funkcji będzie **"tak"**?

```
1. def sprawdz(x):
2.     if x % 2 == 0 and x % 3 != 0:
3.         return "tak"
4.     else:
5.         return "nie"
```

   **a.** Dla liczb podzielnych przez 2 i przez 3.
   **b.** Dla liczb niepodzielnych przez 2 i niepodzielnych przez 3.
   **c.** Dla liczb podzielnych przez 2 i niepodzielnych przez 3.
   **d.** Dla liczb niepodzielnych przez 2 i podzielnych przez 3.

3. Jaką wartość będzie miała zmienna **licznik** po wykonaniu poniższych poleceń?

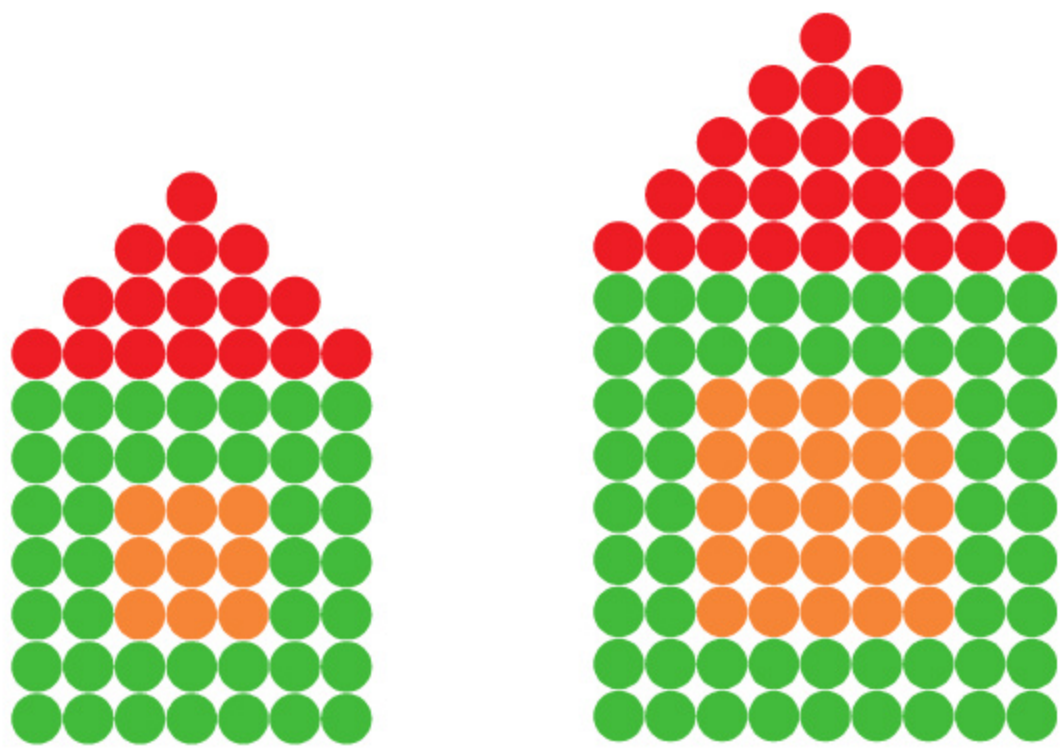
```
1. licznik = 3
2. licznik = licznik + 3
```

   **a.** 3
   **b.** 6
   **c.** 9
   **d.** 12

## ZADANIE DODATKOWE

### Zadanie 1. Trzykolorowe domki

Napisz wzór i zdefiniuj funkcję, która będzie wyliczać liczbę korali danego koloru.



# 16. Wyszukiwanie wzorca w tekście

- Podstawowe operacje na napisach
- Algorytm naiwny wyszukiwania wzorca w tekście
- Porównywanie i przeszukiwanie napisów

## ZADANIE NA START

**Jakie symbole można znaleźć w DNA**

DNA to kwas deoksyrybonukleinowy. Tworzą go nukleotydy zbudowane z deoksyrybozy, jednej grupy fosforanowej oraz jednej zasady azotowej: adeniny (A), guaniny (G), cytozyny (C) lub tyminy (T). Przykładowy łańcuch DNA, który zawiera symbole A, C, G i T, to TCTAAAGATATCGGG. Długość tego łańcucha wynosi 15. Policz, ile razy symbole A, C, G i T występują w łańcuchu TCTAACAGCCCCATATCGGG.



## NAPISY

W języku Python napis to ciąg znaków ujęty w podwójne lub pojedyncze cudzysłowy – zawierający nie tylko litery, lecz także cyfry czy znaki takie jak spacja. Ciąg znaków wewnątrz cudzysłowów jest kopiowany jeden do jednego, co oznacza, że wynikiem polecenia **`print("3 + 4")`** nie jest liczba 7, tylko ciąg znaków **`3 + 4`**.

Aby wypisać napis zdefiniowany jako zmienna, należy zastosować polecenie **`print()`**.

**ZAPAMIĘTAJ!**

```
<zmienna> = "<ciąg znaków>"
print(<zmienna>)
```

Na rys. 1 pokazano przykład definiowania i wyświetlania zmiennych, których wartością są napisy.

```
1. s1 = "biologia"
2. s2 = "molekularna"
3. print(s1)
4. print(s2)
```

**Rys. 1.** Definiowanie i wypisywanie napisów

## DODAWANIE NAPISÓW

Napisy można łączyć ze sobą – jest to tzw. konkatenacja. W Pythonie ciągi znaków dodaje się za pomocą znaku +.

**ZAPAMIĘTAJ!**

```
<zmienna1> = "<ciąg znaków>"
<zmienna2> = "<ciąg znaków>"
print(<zmienna1> + <zmienna2>)
```

W wierszu czwartym na rys. 2 dodawane są trzy napisy, środkowy napis (" ") jest spacją.

```
1. s1 = "biologia"
2. s2 = "molekularna"
3. print(s1 + s2)
4. print(s1 + " " + s2)
```

**Rys. 2.** Konkatenacja napisów

## WYZNACZANIE DŁUGOŚCI NAPISU

Aby wyznaczyć długość ciągu znaków podanego jako argument funkcji, należy wykorzystać funkcję **len()**. Do zliczanych znaków zaliczane są również spacje i znaki przestankowe.

**ZAPAMIĘTAJ!**

```
len(<zmienna>)
```

Na rys. 3 przedstawiono przykład zmiennej **napis**, której przypisano wartość będącą sumą trzech napisów (**s1**, spacja i **s2**), oraz funkcji obliczającej długość tej zmiennej (liczbę znaków).

```
1. s1 = "biologia"
2. s2 = "molekularna"
3. napis = s1 + " " + s2
4. print(len(napis))
```

**Rys. 3.** Wyznaczanie liczby znaków zmiennej **napis**

## ANALIZOWANIE NAPISU

Jeśli trzeba przeanalizować napis znak po znaku, można to zrobić za pomocą pętli **for**.

**ZAPAMIĘTAJ!**

```
for <znak> in <napis>:
    print(<znak>)
```

Na rys. 4 zaprezentowano kod umożliwiający analizę zmiennej **napis**. Efektem jest wypisanie każdej litery przypisanego zmiennej wyrazu w osobnym wierszu.

```
1. napis = "genetyka"
2. for z in napis:
3.     print(z)
```

**Rys. 4.** Wypisywanie kolejnych liter zmiennej **napis**

Pojedyncze znaki napisu można przeanalizować również z wykorzystaniem indeksowania. Wartość indeksu, umieszczona po nazwie zmiennej w nawiasie kwadratowym, wskazuje odpowiedni znak napisu, licząc od zera. Jeśli zmiennej **napis** przypisze się wartość **"genetyka"**, to **napis[0]** będzie oznaczać **"g"**, **napis[4]** będzie oznaczać **"t"**, a **napis[7]** – **"a"**.

W języku Python istnieje również możliwość indeksowania od końca – wtedy zaczyna się od –1.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| g | e | n | e | t | y | k | a |
| –8 | –7 | –6 | –5 | –4 | –3 | –2 | –1 |

W razie potrzeby podczas analizy napisu znak po znaku można połączyć pętlę **for** z funkcją **range** oraz indeksowanie. Pętla jest wykonywana tyle razy, ile znaków ma napis, a indeks zmienia swoje wartości od **0** do liczby (**długość napisu – 1**).

**ZAPAMIĘTAJ!**

```
for <indeks> in range(len(<napis>)):
    print(<napis>[<indeks>])
```

Na rys. 5 pokazano przeglądanie napisu znak po znaku z wykorzystaniem pętli **for**, funkcji **len()** oraz indeksowania.

```
1. napis = "genetyka"
2. for i in range(len(napis)):
3.     print(napis[i])
```

**Rys. 5.** Drugi sposób przeglądania kolejnych liter w zmiennej **napis**

Do sprawdzenia, ile razy w danym napisie pojawia się znak albo ciąg znaków ujęty w cudzysłów, warto wykorzystać metodę **count()**. Wywołuje się ją poprzez napisanie nazwy zmiennej, kropki i nazwy metody.

```
1. napis = "genetyka"
2. print(napis.count("e"))
```

**Rys. 6.** Zliczanie wystąpień znaku **"e"** w zmiennej **napis**

## Ćwiczenie 1. Zliczanie pojedynczych znaków w tekście

Przeanalizuj dane w tabeli i zdefiniuj funkcję **ile(s)**, której parametrem jest ciąg znaków, a wynikiem – lista czterech liczb całkowitych oznaczających, ile razy symbole A, C, G i T występują w tym ciągu znaków.

| Wywołanie funkcji | Wynik |
|---|---|
| `ile("CAATAAAAA")` | `[7, 1, 0, 1]` |
| `ile("TCTAAAGATATCGGG")` | `[5, 2, 4, 4]` |

1. Przeanalizuj treść ćwiczenia i podane przykłady.

- Zlicz wystąpienia kolejnych liter w ciągu znaków **"CAATAAAAA"**: litera A występuje 7 razy, litera C – raz, litera G nie występuje, litera T pojawia się raz.
- Zlicz wystąpienia kolejnych liter w ciągu znaków **"TCTAAAGATATCGGG"**: litera A występuje 5 razy, litera C – 2 razy, litera G – 4 razy, litera T – 4 razy.

2. Przemyśl możliwe rozwiązania problemu, zapisz formuły, a następnie zdefiniuj funkcję.

- Sposób 1 – algorytm wykorzystujący pętlę **for** i instrukcję warunkową.
  - Zapisz kolejne formuły.

  Aby zliczyć wystąpienia znaku A w napisie, należy utworzyć zmienną pomocniczą **ile_A**, która będzie przechowywała liczbę wystąpień symbolu A w danym napisie. Algorytm realizujący to zadanie polega na analizowaniu napisu znak po znaku. W razie napotkania litery A wartość zmiennej **ile_A** jest zwiększana o 1.

```
ile_A += 1
```

Ponieważ to samo dotyczy pozostałych liter, można zastosować instrukcję warunkową. Napis jest złożony wyłącznie z liter ACGT, więc jeśli przetwarzaną literą nie jest A, C ani G, to jest nią litera T.

```
if z == "A":
    ile_A += 1
elif z == "C":
    ile_C += 1
elif z == "G":
    ile_G += 1
else:
    ile_T += 1
```

- Wynikiem funkcji mają być cztery liczby zapisane po przecinku. Taka uporządkowana sekwencja danych umieszczona w nawiasach kwadratowych tworzy strukturę zwaną listą. Aby odwołać się do elementu zawartego w danej liście, należy posłużyć się odpowiadającym mu indeksem. Warto pamiętać, że listy mogą przechowywać różne typy danych, np. dowolne rodzaje liczb, łańcuchy znaków i funkcje, a ponieważ są sekwencjami zmiennymi, można je modyfikować, tj. dodawać do nich elementy lub je usuwać.

```
[ile_A, ile_C, ile_G, ile_T]
```

- Zdefiniuj funkcję **ile(s)**.

```
1. def ile(s):
2.     ile_A = 0; ile_C = 0
3.     ile_G = 0; ile_T = 0
4.     for z in s:
5.         if z == "A":
6.             ile_A += 1
7.         elif z == "C":
8.             ile_C += 1
9.         elif z == "G":
10.            ile_G += 1
11.        else:
12.            ile_T += 1
13.    return [ile_A, ile_C, ile_G, ile_T]
```

- Sposób 2 – algorytm wykorzystujący operacje na napisach.

- Zastosuj metodę **count()** – wykorzystaj metodę **s.count("A")** do obliczenia liczby wystąpień symbolu A, **s.count("C")** – do obliczenia liczby wystąpień symbolu C itd.

- Zdefiniuj funkcję **ile(s)**.

```
1. def ile(s):
2.     return [s.count("A"), s.count("C"), s.count("G"), s.count("T")]
```

3. Sprawdź działanie skryptu z różnymi danymi, w tym z napisami złożonymi z jednakowych liter.

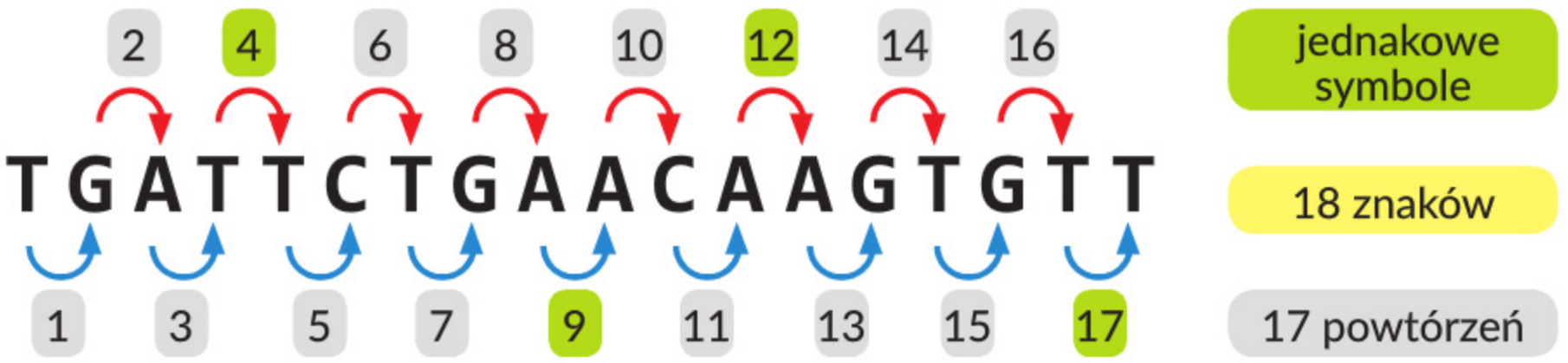
## Ćwiczenie 2. Zliczanie ciągów znaków w tekście

Przeanalizuj dane w tabeli i zdefiniuj funkcję **pary(s)**, której parametrem jest łańcuch DNA, a wynikiem – liczba wystąpień par jednakowych symboli (np. TTAT – 1, TTTA – 2, TATA – 0).

| Wywołanie funkcji | Wynik |
|---|---|
| `pary("TGATTCTGAACAAGTGTT")` | 4 |
| `pary("TGATATTCGATGTGAAAAAGTCATACTGTT")` | 6 |

1. Przeanalizuj treść ćwiczenia i podane przykłady.

- Wykonaj rysunek pomocniczy.




- Aby wyszukać jednakowe symbole, trzeba przeanalizować cały napis znak po znaku. Porównuje się pary, czyli bieżący znak z następnym, np. TG, GA, AT, TT itd. Pętla będzie wykonywana *x* razy, gdzie *x* to **(długość napisu – 1)**.

2. Zapisz kolejne formuły – przy konstruowaniu warunku zastosuj indeksowanie. Sprawdź, czy znak o indeksie `i` jest równy znakowi o indeksie `i + 1`.

```
s[i] == s[i + 1]
```

3. Zdefiniuj funkcję **pary(s)**.

```
1. def pary(s):
2.     ile = 0
3.     for i in range(len(s) - 1):
4.         if s[i] == s[i + 1]:
5.             ile += 1
6.     return ile
```

4. Sprawdź działanie skryptu z różnymi danymi.

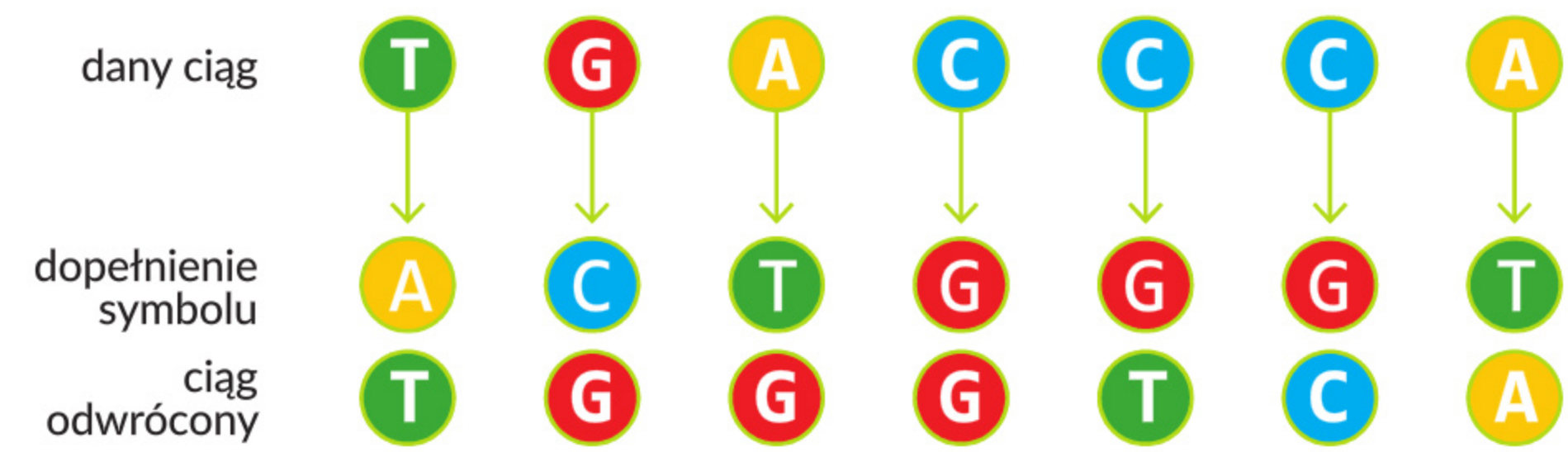
## Ćwiczenie 3. Zmiana struktury ciągu znaków

Odwrotne uzupełnienie łańcucha DNA to ciąg utworzony przez odwrócenie danego ciągu i dopełnienie każdego symbolu, tj. zamiany A na T i odwrotnie oraz C na G i odwrotnie (np. CTGA → AGTC → TCAG). Przeanalizuj dane w tabeli i zdefiniuj funkcję **oduz(s)**, której parametrem jest łańcuch DNA, a wynikiem jego odwrotne uzupełnienie.

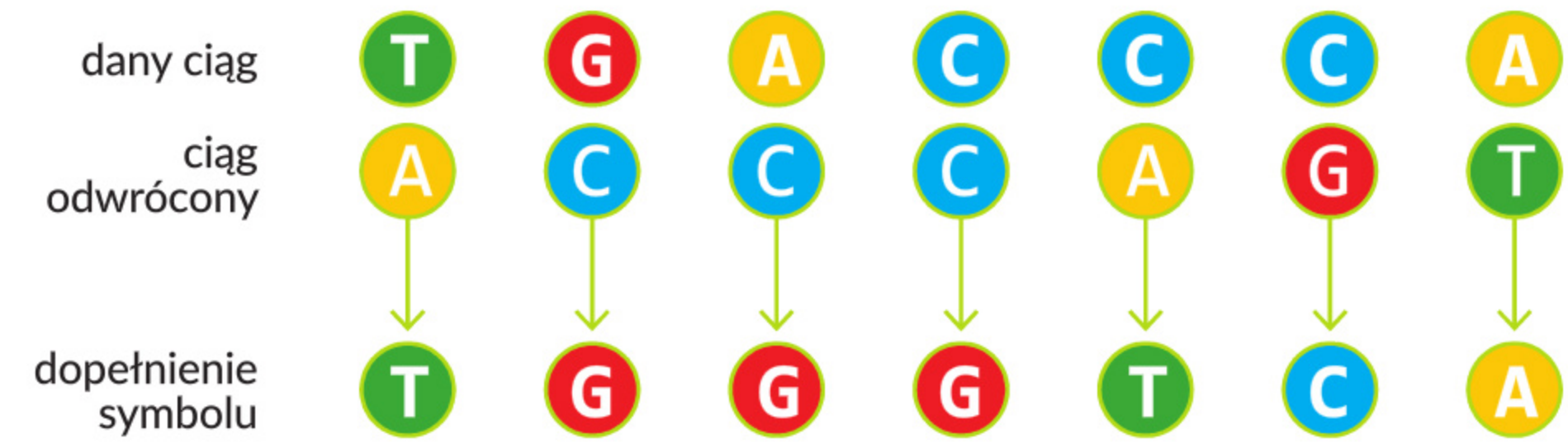
| Wywołanie funkcji | Wynik |
|---|---|
| **oduz("TGACCCA")** | **"TGGGTCA"** |
| **oduz("TGATATTCGATGTGAAAAA")** | **"TTTTTCACATCGAATATCA"** |

1. Przeanalizuj treść ćwiczenia i podane przykłady.

- Dla danego ciągu TGACCCA trzeba utworzyć jego dopełnienie (zamiana litery A na T, T na A, C na G i G na C) – wynikiem będzie ACTGGGT. Następnie trzeba go odwrócić, czyli zapisać od końca – wynikiem będzie TGGGTCA. Wykonaj rysunek pomocniczy.




- Kolejność operacji (zamiana i odwracanie) nie ma znaczenia.




2. Zapisz kolejne formuły.

- Najpierw nadaj wartość początkową zmiennej **t**, która będzie wynikiem. Wynik ma być napisem, więc wartość początkowa będzie napisem pustym **("")**.

```
t = ""
```

- Aby w Pythonie zmodyfikować napis – zmienić lub zastąpić w nim poszczególne znaki – utwórz zmienną do przechowywania przekształconego łańcucha DNA, np. **zn**, odwróć ciąg i zamień parami symbole, z uwzględnieniem kolejności napisów w dodawaniu.

```
if s[i] == "A":
    zn = "T"
elif s[i] == "T":
    zn = "A"
elif s[i] == "C":
    zn = "G"
else:
    zn = "C"
```

- Algorytm polega na przeglądaniu danego napisu. W trakcie przeglądania, gdy kolejnym symbolem będzie A, to zmiennej pomocniczej **zn** algorytm nada wartość **T** – i odwrotnie; gdy symbolem będzie C, to zmiennej pomocniczej **zn** nada wartość **G** – i odwrotnie. Do wyniku algorytm doda wartość zmiennej pomocniczej **zn** – w ten sposób odwróci przekształcany napis.

```
t = zn + t
```

3. Zdefiniuj funkcję **oduz(s)**.

```
1. def oduz(s):
2.     t = ""
3.     for i in range(len(s)):
4.         if s[i] == "A":
5.             zn = "T"
6.         elif s[i] == "T":
7.             zn = "A"
8.         elif s[i] == "C":
9.             zn = "G"
10.        else:
11.            zn = "C"
12.        t = zn + t
13.    return t
```

4. Sprawdź działanie skryptu z różnymi danymi.

## Ćwiczenie 4. Zmiana struktury ciągu znaków

Przeanalizuj dane w tabeli i zdefiniuj funkcję **ta(s)**, której parametrem jest łańcuch DNA, a wynikiem – łańcuch DNA z gwiazdką (*) w miejscu symboli C, G i T.

| Wywołanie funkcji | Wynik |
|---|---|
| `ta("TGACCCA")` | `"**A***A"` |
| `ta("TGATATTCGATGTGAAAAA")` | `"**A*A****A****AAAAA"` |

# ALGORYTM

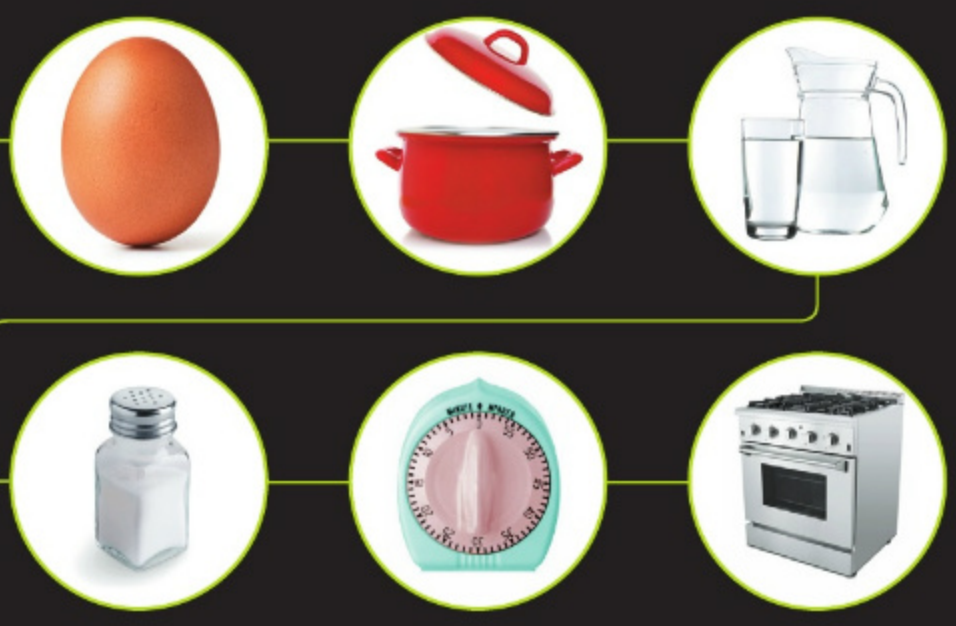
**Algorytm to przepis postępowania prowadzący do rozwiązania problemu. Określa dane wejściowe oraz ciąg czynności, które należy w tym celu wykonać. Wykonawcami algorytmu mogą być i komputer, i człowiek.**

**1.** **Określenie problemu:**
zwięzłe, precyzyjne

Chcę zjeść jajko na miękko na śniadanie.

**2.** **Zdefiniowanie modeli i pojęć:**
opisanie danych wejściowych i zależności wyniku od danych

Co jest potrzebne?



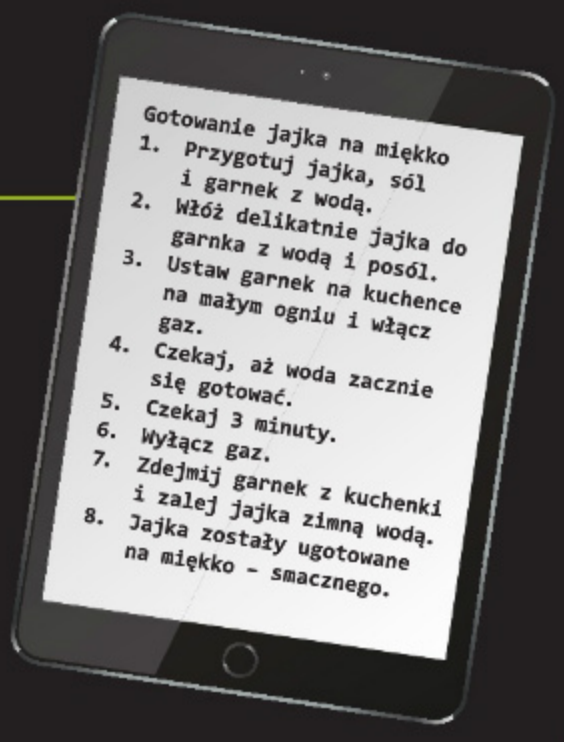
**3.** **Znalezienie rozwiązania:**
opisanie możliwych rozwiązań i wybór najlepszego

Metody gotowania



**4.** **Zaprogramowanie rozwiązania:**
napisanie kodu programu

Przepis



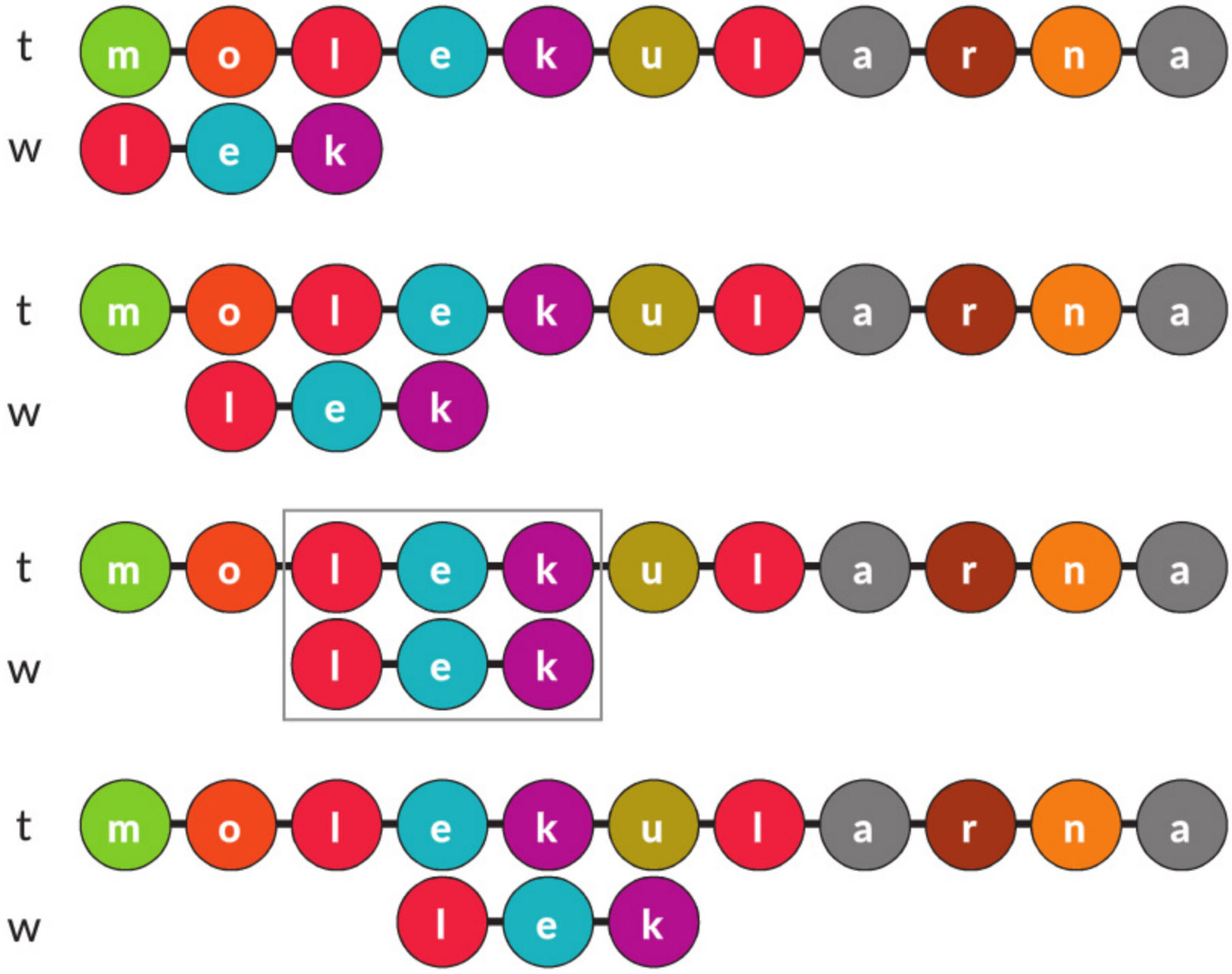

**5.** **Testowanie rozwiązania:**
dla różnych danych spełniających przyjęte założenia

Śniadanie



## ALGORYTM NAIWNY WYSZUKIWANIA WZORCA W TEKŚCIE

Wzorzec to spójny ciąg znaków, który występuje w przeszukiwanym tekście – jest nim np. fraza **lek** występująca w słowie **molekularna**. Algorytmy umożliwiające wyszukiwanie wzorca w danym skończonym łańcuchu znaków są wykorzystywane m.in. w edytorach tekstu, przeglądarkach internetowych oraz programach pocztowych. Za pomocą wzorców można też określić, z czego ma się składać adres e-mail. Sekwencja znaków zadana do odszukania może się powtarzać albo w ogóle nie występować w analizowanym tekście.




**Rys. 7.** Cztery z dziewięciu etapów procesu wyszukiwania wzorca w słowie **molekularna**

Na powyższym rysunku wzorzec został odnaleziony na trzecim etapie procesu – wszystkie trzy litery są zgodne. Jeżeli chcesz znaleźć wszystkie wystąpienia wzorca w tekście, to kontynuuj przeszukiwanie tekstu od następnego znaku, aż do momentu, gdy liczba nieprzebadanych znaków stanie się mniejsza od długości wzorca.

**ZAPAMIĘTAJ!**

**Algorytm naiwny wyszukiwania wzorca** (inaczej – algorytm N) to najprostszy algorytm, który polega na porównywaniu kolejnych liter tekstu (**t**) i wzorca (**w**), począwszy od pierwszej litery każdego z nich. W razie niezgodności wzorzec przesuwa się o jedną pozycję w prawo i algorytm bada drugi znak tekstu i pierwszy znak wzorca. Jeśli znaki są zgodne, porównywane są następne znaki tekstu i wzorca.

**W PRAKTYCE**

W edytorach tekstu można wyszukiwać konkretne napisy lub posłużyć się symbolami wieloznacznymi, np. użyć symbolu gwiazdki (*) do wyszukania ciągu znaków. Do wzorca b*a pasują takie wyrazy, jak biologia, biochemiczna, biofizyka, białka.

## Ćwiczenie 5. Przeszukiwanie DNA

Dany jest łańcuch DNA o sekwencji TCTAACAGCCCCATATCGGG. Na którym etapie algorytmu N zostanie odnaleziony wzorzec CAT? Ile wzorców zostanie znalezionych?

## PODSUMOWANIE

- Algorytmika i programowanie mają zastosowanie w różnych dziedzinach nauki, m.in. w biologii molekularnej, a także w codziennym funkcjonowaniu w społeczeństwie cyfrowym.
- Napis, czyli ciąg znaków zdefiniowany jako zmienna i ujęty w cudzysłowy, wyświetla się za pomocą polecenia **`print`**.
- Konkatenację (łączenie) napisów umożliwia znak dodawania.
- Długość ciągu znaków wyznacza się za pomocą funkcji **`len()`**.
- Aby przeanalizować napis znak po znaku, należy zastosować pętlę **`for`** albo pętlę **`for`** z funkcją **`range`** i indeksowanie.
- Napisy mają wiele wbudowanych metod, które można wykorzystać do rozwiązywania problemów, np. **`count()`** pozwala sprawdzić, ile razy w napisie powtarza się dany znak lub ciąg znaków. Metodę wywołuje się przez napisanie nazwy zmiennej, kropki i nazwy metody.
- Chociaż w rozbudowanych systemach stosuje się bardziej zaawansowane algorytmy wyszukiwania, które działają szybciej, do prostych zastosowań wystarczy algorytm naiwny.
- Algorytm naiwny wyszukiwania wzorca w tekście polega na porównywaniu kolejnych liter tekstu i wzorca, począwszy od pierwszej litery każdego z nich. W razie niezgodności wzorzec przesuwa się o jedną pozycję w prawo i algorytm bada drugi znak tekstu oraz pierwszy znak wzorca. Jeśli znaki są zgodne, porównywane są następne znaki.

## PYTANIA SPRAWDZAJĄCE

1. Jakie podstawowe operacje można wykonać na napisach?
2. W jaki sposób można w Pythonie wyświetlić znak po znaku? Podaj dwa sposoby.
3. Ile razy trzeba porównywać dwa znaki, aby znaleźć w tekście o długości **`n`** wzorzec o długości **`w`**?

## PYTANIA QUIZOWE

1. Jaka wartość zostanie wypisana dla napisu `w = "wiaderko"` w wyniku działania polecenia `print(len(w))`?
   **a.** 5
   **b.** 10
   **c.** 12
   **d.** 8
2. Jaka litera zostanie wypisana dla napisu `w = "kaktus"` w wyniku działania polecenia `print(w[-1])`?
   **a.** k
   **b.** s
   **c.** u
   **d.** a
3. Ile razy wykona się poniższa pętla dla napisu `w = "laseczek"`?

```
for z in w:
    print(z)
```

   **a.** 8
   **b.** 10
   **c.** 2
   **d.** 7

## ZADANIA DODATKOWE

### Zadanie 1. Sekwencja błędów

Odległość Hamminga między dwoma łańcuchami DNA oznacza liczbę symboli, którymi różnią się oba łańcuchy. Odległość Hamminga między łańcuchami `s` i `t` o tej samej długości wynosi 8. Niepasujące symbole są zaznaczone kolorem:

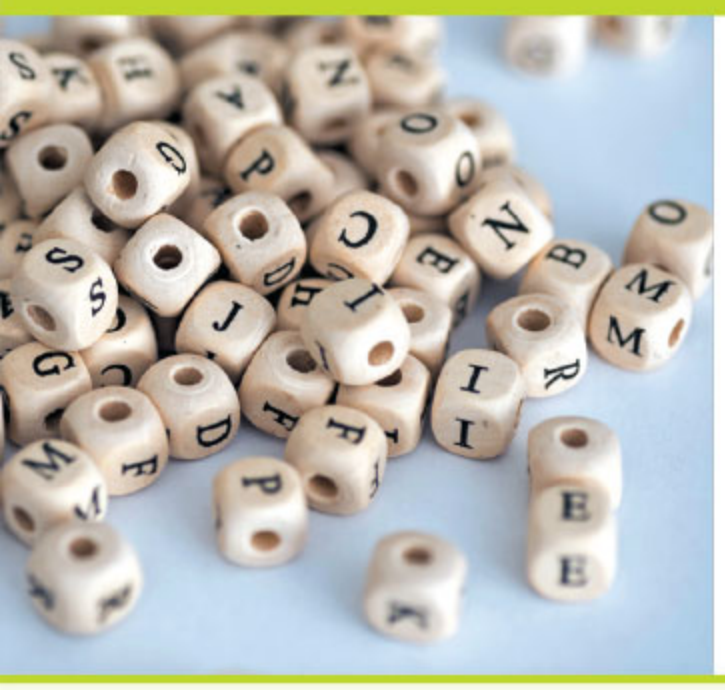
- łańcuch `s` – CAGGCTACTACCGGTAT,
- łańcuch `t` – CATCGTAATGCAGGGCT.

Przeanalizuj dane w tabeli i zdefiniuj funkcję `hamming(s, t)`, gdzie `s` i `t` to łańcuchy o tej samej długości. Wynik funkcji to liczba będąca odległością Hamminga między danymi łańcuchami.

| Wywołanie funkcji | Wynik |
|---|---|
| `hamming("CAGGTT", "CAGGTT")` | `0` |
| `hamming("CAGGCTACTACCGGTAT", "CATCGTAATGCAGGGCT")` | `8` |

### Zadanie 2. Jak wyszukać tekst?

Sprawdź, jak wyszukiwać napisy za pomocą symboli wieloznacznych w edytorze tekstu. Podaj kilka przykładów wyszukiwań.



# 17. Przetwarzanie napisów

- Wyodrębnianie fragmentu napisu
- Szyfry przestawieniowe
- Sprawdzanie, czy napis jest palindromem

## ZADANIE NA START

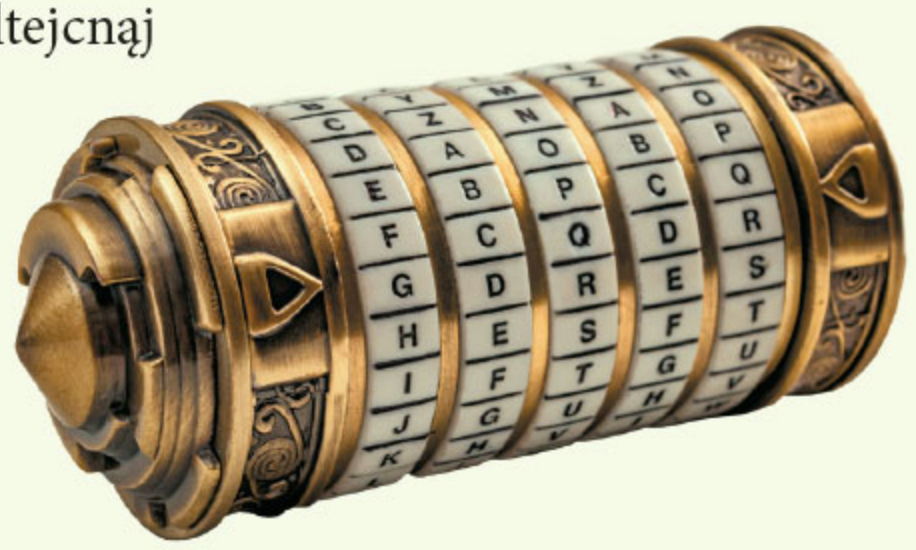
**Czy umiesz odczytać poniższe sentencje**

Odczytaj poniższe zdania, a następnie opisz metodę szyfrowania, jaką zastosowano w każdym przypadku.

patetyr śoin e wimra ęejzdneai
tytepa einśor w ęraim ainezdej
aqpdebtayity raoośynoiuel wy meixabrlęt jvewdfzgeinpiaaq
apeytt roinśe w mraię jndeieza

dgiz erdawr bąąit maw óiyrl ceą
eizdg awrd ąibąr mat yróiw ącel
godczaiseg dwrownad rląwbqiyąx tsaqmj wmicópriyh ltejcnąj
gidze dwra rbiąą tam wóriy lecą

eb zrpca yin eamk łocayz
zeb ycarp ein am yzcałok
bveuzm pqriapcoyl nbipek mhau kcoałlahctzmyz
bez parcy nie ma kłczoay



## WYODRĘBNIANIE FRAGMENTU NAPISU

Podczas przetwarzania napisów można odwoływać się do poszczególnych znaków za pomocą indeksów. Umożliwia to wyodrębnianie fragmentów napisów. Indeksy to liczby z zakresu od **0** do **(długość napisu - 1)**. Za pomocą nawiasów kwadratowych uzyskuje się dostęp do pojedynczego znaku – wystarczy podać jego indeks.

```
1. s = "szyfrowanie"
2. print(s[0])    # s
3. print(s[9])    # i
```

**Rys. 1.** Wypisanie znaków o indeksie 0 i 9

> **ZAPAMIĘTAJ!**
>
> W Pythonie jednowierszowy **komentarz** rozpoczyna się od znaku #. Komentarze wielowierszowe są zawarte między trzema cudzysłowami lub apostrofami.

Jeżeli w nawiasie kwadratowym poda się dwie liczby oddzielone dwukropkiem, uzyska się fragment napisu, zaczynając od indeksu, który jest pierwszą liczbą, a kończąc na indeksie o **1** mniejszym niż druga liczba. Jeśli przed dwukropkiem nie będzie liczby, to domyślnie pierwsza liczba będzie równa **0**.

> **ZAPAMIĘTAJ!**
>
> Instrukcja **<napis>[<liczba1> : <liczba2>]** wyodrębni **<napis>** od indeksu **<liczba1>** do indeksu **<liczba2> - 1.**

Na rys. 2 pokazano, jak używać indeksów do wywołania poszczególnych części danego ciągu znaków.

```
1. s = "szyfrowanie"
2. print(s[3:7])    # frow
3. print(s[0:5])    # szyfr
4. print(s[:5])     # szyfr
```

**Rys. 2.** Wypisanie fragmentu napisu **s**

W nawiasach można też podać liczby ujemne. Dzięki temu znaki będą liczone od końca: **–1** to wówczas ostatni znak, **–2** przedostatni itd.

```
1. s = "szyfrowanie"
2. print(s[-1])       # e
3. print(s[-5:-2])    # wan
4. print(s[-5:])      # wanie
5. print(s[:-2])      # szyfrowan
```

**Rys. 3.** Wypisanie fragmentu napisu **s** za pomocą indeksów ujemnych

W prosty sposób można uzyskać napis od końca. Wystarczy podać w nawiasie kwadratowym po dwóch dwukropkach indeks **–1**.

```
1. s = "szyfrowanie"
2. print(s)         # szyfrowanie
3. print(s[::])     # szyfrowanie
4. print(s[::-1])   # einaworfyzs
```

**Rys. 4.** Wypisanie napisu normalnie i wspak

## SZYFRY PRZESTAWIENIOWE

Kodowanie wiadomości może się odbyć przez przestawienie kolejnych par znaków. W tak zaszyfrowanym tekście występują wszystkie znaki z tekstu jawnego, ale w innej kolejności. Kolejność liter zmieniana jest według określonego schematu, np. pisanie wspak.

**ZAPAMIĘTAJ!**

**Szyfrogram** to tekst, który został zaszyfrowany.

**Tekst jawny** to tekst, który nie został zaszyfrowany lub został odszyfrowany.

Nauka, która zajmuje się szyfrowaniem i deszyfrowaniem wiadomości, nosi nazwę **kryptografii**.

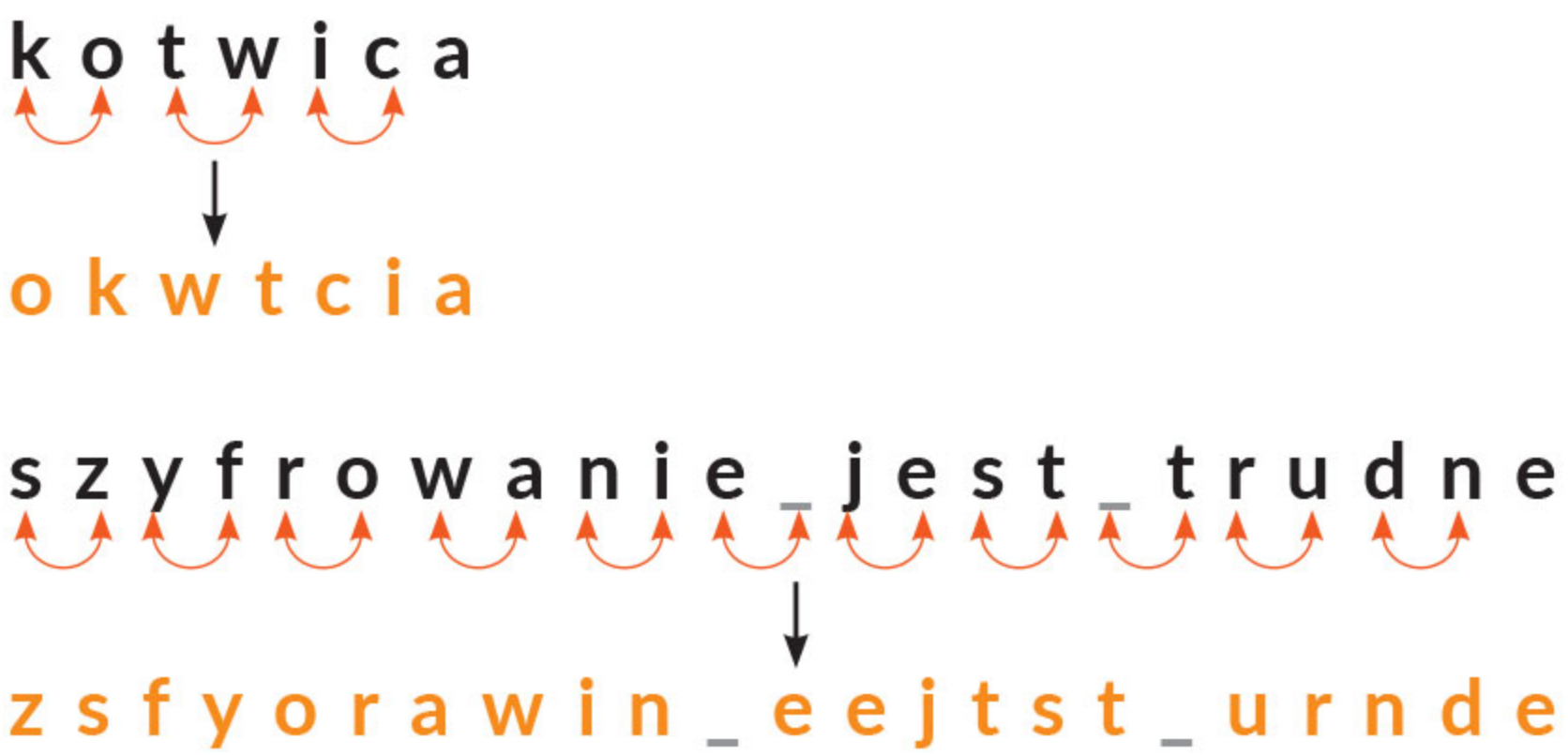
### Ćwiczenie 1. Przestawianie parami

Przeanalizuj dane w tabeli i zdefiniuj funkcję **`szyfr(s)`**, której parametrem jest tekst jawny składający się z małych liter alfabetu łacińskiego i pojedynczych spacji, a wynikiem – szyfrogram utworzony przez przestawienie kolejnych par znaków, przy czym spacja jest traktowana jak każda inna litera i również podlega zamianie.

| Wywołanie funkcji | Wynik |
|---|---|
| `szyfr("kotwica")` | `"okwtcia"` |
| `szyfr("szyfrowanie jest trudne")` | `"zsfyorawin eejtst urnde"` |

1. Przeanalizuj treść ćwiczenia i podane przykłady.

- Rozrysuj algorytm krok po kroku.




- Do zmiennej **`t`** jest przypisany pusty napis. W każdym obrocie pętli rozpatrywane są dwa kolejne znaki, a dokładniej: zamieniana jest ich kolejność i po tej zamianie są one dopisywane na koniec zmiennej **`t`**. Gdy napis ma długość nieparzystą, dopisywany jest ostatni znak.

2. Zapisz kolejne formuły.

- Zastosuj pętlę **for**, aby przeglądać dany tekst co dwa znaki i dopisywać do zmiennej pary znaków zamienione kolejnością w stosunku do danych.

```
for i in range(0, len(s) - 1, 2):
    t = t + s[i + 1] + s[i]
```

- Dopisanie ostatniego znaku to warunek wykonywany po zakończeniu pętli.

```
if len(s) % 2 != 0:
    t = t + s[len(s) - 1]
```

3. Zdefiniuj funkcję **szyfr(s)**.

```
1. def szyfr(s):
2.     t = ""
3.     for i in range(0, len(s) - 1, 2):
4.         t = t + s[i + 1] + s[i]
5.     if len(s) % 2 != 0:
6.         t = t + s[len(s) - 1]
7.     return t
```

4. Sprawdź działanie skryptu z różnymi danymi, w tym – z napisami o parzystej i nieparzystej długości.

- Dlaczego ważne jest, żeby testować funkcję z napisami zarówno parzystej, jak i nieparzystej długości?

**PARKAN**

Choć *Encyklopedia Britannica* opisuje parkan jako szyfr popularny w szkołach, to osoba niemająca z nim wcześniej do czynienia musi się sporo natrudzić, zanim odczyta ukrytą wiadomość. Prosta wersja szyfru polega na zapisaniu liter tekstu jawnego w dwóch wierszach. Na początek usuwa się wszystkie spacje – to częsta praktyka w kryptografii, ponieważ podział na wyrazy sprawia, że złamanie szyfru staje się łatwiejsze. Następnie należy w pierwszym wierszu zapisać litery na pozycjach nieparzystych, w drugim – na pozycjach parzystych. Aby utworzyć szyfrogram, trzeba odczytać tekst wierszami. Oczywiście, przy szyfrowaniu parkanem nie trzeba ograniczać się do dwóch rzędów.

| Tekst jawny | Szyfrowanie | Szyfrogram |
|---|---|---|
| albumy | a b m<br>l u y | abmluy |
| domek | d m k<br>o e | dmkoe |

**Rys. 5.** Szyfrowanie parkanem

## Ćwiczenie 2. Parkan

Przeanalizuj dane w tabeli i zdefiniuj funkcję **parkan(s)**, której parametrem jest tekst jawny złożony z małych liter alfabetu łacińskiego, a wynikiem – szyfrogram utworzony za pomocą szyfru parkan.

| Wywołanie funkcji | Wynik |
|---|---|
| `parkan("fronty")` | `"fotrny"` |
| `parkan("tajne")` | `"tjean"` |

1. Przeanalizuj treść ćwiczenia i podane przykłady.

- Wykonaj rysunki pomocnicze.




2. Opracuj algorytm – do pustego słowa najpierw trzeba dopisać znaki zaznaczone kolorem pomarańczowym, a potem – zielonym.

3. Zdefiniuj funkcję **parkan(s)**.

```
1. def parkan(s):
2.     w = ""
3.     for i in range(0, len(s), 2):
4.         w = w + s[i]
5.     for i in range(1, len(s), 2):
6.         w = w + s[i]
7.     return w
```

4. Sprawdź działanie skryptu z różnymi danymi.

## Ćwiczenie 3. Parkan bis

Zaproponuj inne rozwiązanie ćwiczenia 2.

### PALINDROM

Słowa, zdania i liczby identyczne niezależnie od tego, czy są czytane od lewej do prawej, czy od prawej do lewej, np. kajak, „A to kanapa pana kota", 18081, XIX, to palindromy (z gr. biec z powrotem). Wyrażenia tego typu tworzono już w starożytności. Nie wiadomo, czy stanowiło to tylko zabawę, czy też element obrządków magicznych albo rodzaj kodu. Klasyczna metoda sprawdzania, czy dane wyrażenie jest palindromem, polega na porównywaniu pierwszego znaku z ostatnim, drugiego z przedostatnim itd.

# SCHEMATY BLOKOWE

**Przedstawienie algorytmu w postaci schematu blokowego może czasami ułatwić analizę i zaprogramowanie jego działania. Poniżej przedstawiono schemat blokowy algorytmu sprawdzania, czy podane słowo jest palindromem.**



początek algorytmu

wprowadzenie danych

START

podaj s

wykonywane działania

i = 0

*nie*

czy
i == len(s//2)?

zwiększ
i o 1

*nie*

czy
s[i] != s[-i-1]?

sprawdzanie warunku (kierunek kolejnego kroku jest zgodny z wynikiem warunku)

*tak*

*tak*

wynikiem
jest prawda

wynikiem
jest fałsz

wyprowadzanie wyników

STOP

STOP

koniec algorytmu

Wobec tego w przypadku wyrażenia złożonego z siedmiu znaków wykonuje się trzy porównania (tak jak w przypadku wyrażenia złożonego z sześciu znaków), gdyż środkowego znaku nie ma potrzeby porównywać.

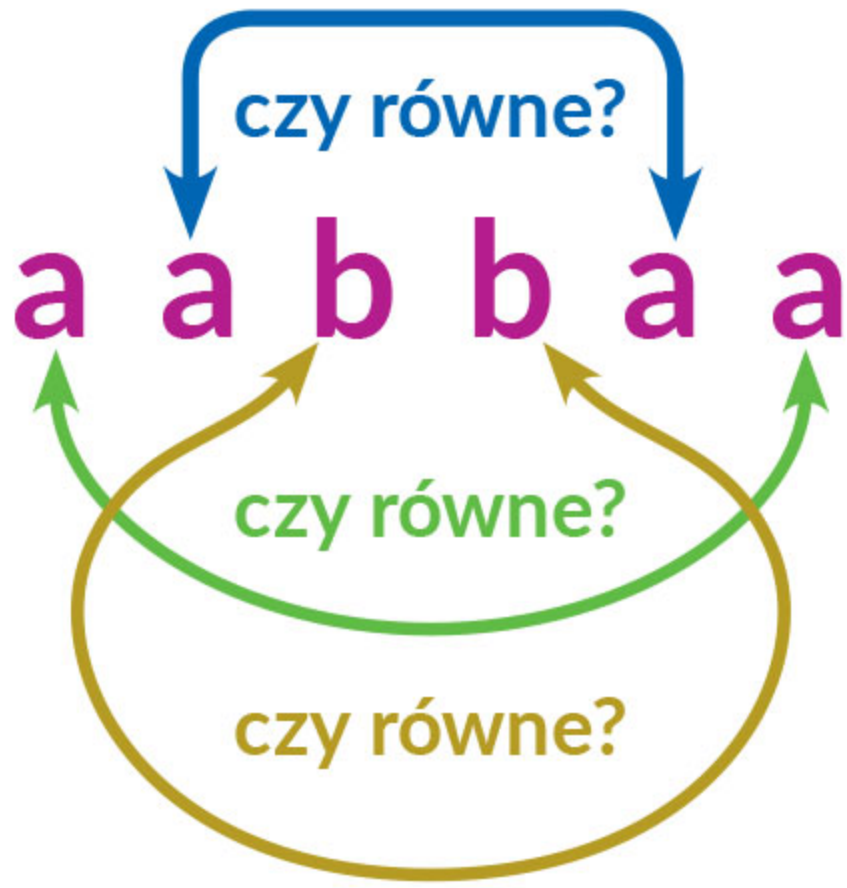
## Ćwiczenie 4. Czy to jest palindrom

Przeanalizuj dane w tabeli i zdefiniuj funkcję logiczną **`czy_palindrom(s)`**, której parametrem jest napis zawierający tylko małe litery alfabetu łacińskiego, a wynikiem – wartość logiczna **`True`**, gdy napis jest palindromem, lub **`False`**, gdy nim nie jest.

| Wywołanie funkcji | Wynik |
|---|---|
| `czy_palindrom("aabbaa")` | `True` |
| `czy_palindrom("aaxyaa")` | `False` |

1. Przeanalizuj treść ćwiczenia i podane przykłady.

- Wykonaj rysunek pomocniczy lub schemat blokowy, jak na poprzedniej stronie.




2. Opracuj algorytm, który polega na sprawdzeniu znaków parami, czyli najpierw bada, czy pierwsza litera jest identyczna z ostatnią, a jeśli tak, to kontynuuje sprawdzanie kolejnych dwóch liter (drugiej i przedostatniej itd.). Jeśli litery nie są identyczne, kończy działanie funkcji i otrzymuje w wyniku **`False`**.

3. Zdefiniuj funkcję **`czy_palindrom(s)`**.

```
1. def czy_palindrom(s):
2.     for i in range(len(s) // 2):
3.         if s[i] != s[-i - 1]:
4.             return False
5.     return True
```

4. Sprawdź działanie funkcji z różnymi danymi, zwłaszcza z napisami, które są palindromami.

## Ćwiczenie 5. Palindrom i odwracanie napisów

Rozwiąż ćwiczenie 4 z wykorzystaniem odwracania słowa.

## Ćwiczenie 6. Czy zdanie jest palindromem

Przeanalizuj dane w tabeli i zdefiniuj funkcję logiczną **`czy_palindrom2(s)`**, której parametrem jest napis zawierający małe litery alfabetu łacińskiego i spacje, a wynikiem wartość logiczna **`True`**, gdy napis jest palindromem, lub **`False`**, gdy nim nie jest.

| Wywołanie funkcji | Wynik |
|---|---|
| `czy_palindrom2("a to kanapa pana kota")` | `True` |
| `czy_palindrom2("a to kanapka pana kota")` | `False` |

1. Przeanalizuj treść ćwiczenia i podane przykłady.

- Zauważ, że ćwiczenie można rozwiązać w dwóch krokach – najpierw pomiń spacje, potem zastosuj funkcję z ćwiczenia 4.

2. Zdefiniuj funkcję **`pomin(s)`**, która usuwa spacje z tekstu.

```
1. def pomin(s):
2.     pom = ""
3.     for zn in s:
4.         if zn != " ":
5.             pom = pom + zn
6.     return pom
```

3. Zdefiniuj funkcję **`czy_palindrom2(s)`** z wykorzystaniem funkcji **`czy_palindrom(s)`**.

```
1. def czy_palindrom2(s):
2.     return czy_palindrom(pomin(s))
```

4. Sprawdź działanie skryptu z różnymi danymi, w tym z napisami zawierającymi spacje.

### PRAKTYCZNE WYKORZYSTANIE SZYFROWANIA

Zabezpieczanie danych przez szyfrowanie to jedno z większych wyzwań w społeczeństwie cyfrowym. Szyfrować można m.in. pojedyncze pliki, foldery, partycje dysków i inne nośniki danych, e-maile albo transmisje danych. Możliwość szyfrowania oferują liczne aplikacje, a także popularne programy do kompresowania danych, takie jak 7-Zip czy WinRAR. Dzięki wykorzystaniu metod informatycznych wiele szyfrów można złamać, ale i tak odczytanie zaszyfrowanej wiadomości może sprawić trudność przeciętnemu człowiekowi.

**POSZUKAJ W INTERNECIE**

Wejdź na stronę **https://pl.khanacademy.org**, wybierz *Podróż w krainę kryptografii* i sprawdź, jak ludzie chronili informacje kiedyś i jak robią to dziś.

**WIEM WIĘCEJ**

Anagram to wyraz, wyrażenie lub zdanie powstałe przez przestawienie liter albo sylab innego wyrazu, wyrażenia lub zdania. W 1609 roku Galileusz odkrył pierścienie Saturna. Aby zabezpieczyć się przed wścibstwem żądnej sławy konkurencji, włoski astronom zanotował informację o swojej obserwacji za pomocą anagramu: SMAISMRMILMEPOETALEVMIBUNENUGTTAVIRAS. Oznaczało to: *Altissimum planetam tergeminum observavi* (Najwyższą planetę obserwowałem jako potrójną).

## PODSUMOWANIE

- W Pythonie można operować fragmentami napisów. Stosowanie indeksów pozwala na wyodrębnienie pojedynczego znaku i całego fragmentu, a nawet na odwrócenie napisu.
- Szyfrogram to tekst, który został zaszyfrowany. Tekst jawny to tekst, który nie został zaszyfrowany lub został odszyfrowany.
- Szyfry przestawieniowe polegają na zamianie kolejności znaków, długość napisu wówczas się nie zmienia.
- Szyfrowanie parkanem polega na zapisaniu znaków tekstu jawnego w wierszach. Po usunięciu spacji należy w pierwszym wierszu zapisać pierwszy znak, w drugim drugi, w trzecim trzeci itd., a następnie przejść do kolejnych rzędów i na koniec spisać znaki z kolejnych wierszy jako poszczególne słowa. Osoba chcąca odszyfrować tekst musi odczytać najpierw pierwszy znak pierwszego słowa, potem pierwszy znak drugiego słowa itd.
- Palindrom to słowo lub zdanie identyczne niezależnie od tego, czy jest czytane od lewej do prawej, czy od prawej do lewej. Sprawdzamy, czy napis jest palindromem, przez porównanie kolejno par znaków: pierwszej i ostatniej, drugiej i przedostatniej itd. Jeżeli napotkamy różne znaki, to już wiemy, że napis nie jest palindromem.
- Zabezpieczanie danych przez szyfrowanie jest nieodzowne w społeczeństwie cyfrowym. Szyfrować można m.in. pojedyncze pliki, foldery, partycje dysków i inne nośniki, e-maile oraz transmisje danych.

## PYTANIA SPRAWDZAJĄCE

1. Jak napisać pierwszy znak danego napisu? A jak – ostatni?
2. W jaki sposób można napisać odwrócony napis?
3. Omówiony w ćwiczeniu 4 klasyczny sposób sprawdzania, czy napis jest palindromem, w przypadku napisu 10-znakowego uruchomi pętlę pięć razy. Ile razy zostanie wywołana pętla, gdy napis ma 11 znaków?

## PYTANIA QUIZOWE

1. Jaki napis zostanie wypisany dla napisu `w = "kolumna"` w wyniku działania polecenia `print(w[0:5])`?
   **a.** kolumna
   **b.** kolumn
   **c.** kolum
   **d.** anmulok
2. Jakiego polecenia należy użyć, aby dla napisu `w = "szyfrant"` wypisać 4 ostatnie znaki?
   **a.** `print(w[-4:1])`
   **b.** `print(w[-4:])`
   **c.** `print(w[:-1])`
   **d.** `print(w[4:-1])`
3. Jakim znakiem należy zacząć jednowierszowy komentarz?
   **a.** `:`
   **b.** `;`
   **c.** `#`
   **d.** `$`

## ZADANIA DODATKOWE

### Zadanie 1. Ukrywanie tekstu

Przeanalizuj dane w tabeli i zdefiniuj funkcję **`ukryj(s)`**, której parametrem jest napis złożony z małych liter alfabetu łacińskiego, a wynikiem – napis ze wstawioną co drugą literą wybraną losowo.

| Wywołanie funkcji | Wynik |
|---|---|
| `ukryj("zima")` | `"zkitmwau"` |
| `ukryj("alamakota")` | `"axldaamaackvostvac"` |

**Wskazówka:** do losowania wykorzystaj funkcję **`randint()`** z modułu **`random`**.

### Zadanie 2. ROT13

Zapoznaj się z informacją na temat szyfru ROT13 na s. 199, a następnie przeanalizuj dane w tabeli i zdefiniuj funkcję **`rot(s)`**, której parametrem jest napis zawierający wielkie litery alfabetu łacińskiego, a wynikiem – napis zakodowany według podanej zasady.

| Wywołanie funkcji | Wynik |
|---|---|
| `rot("GRAT")` | `"TENG"` |
| `rot("INFORMACJA")` | `"VASBEZNPWN"` |

# , CZYLI SZYFRY

## ALFABET MORSE'A

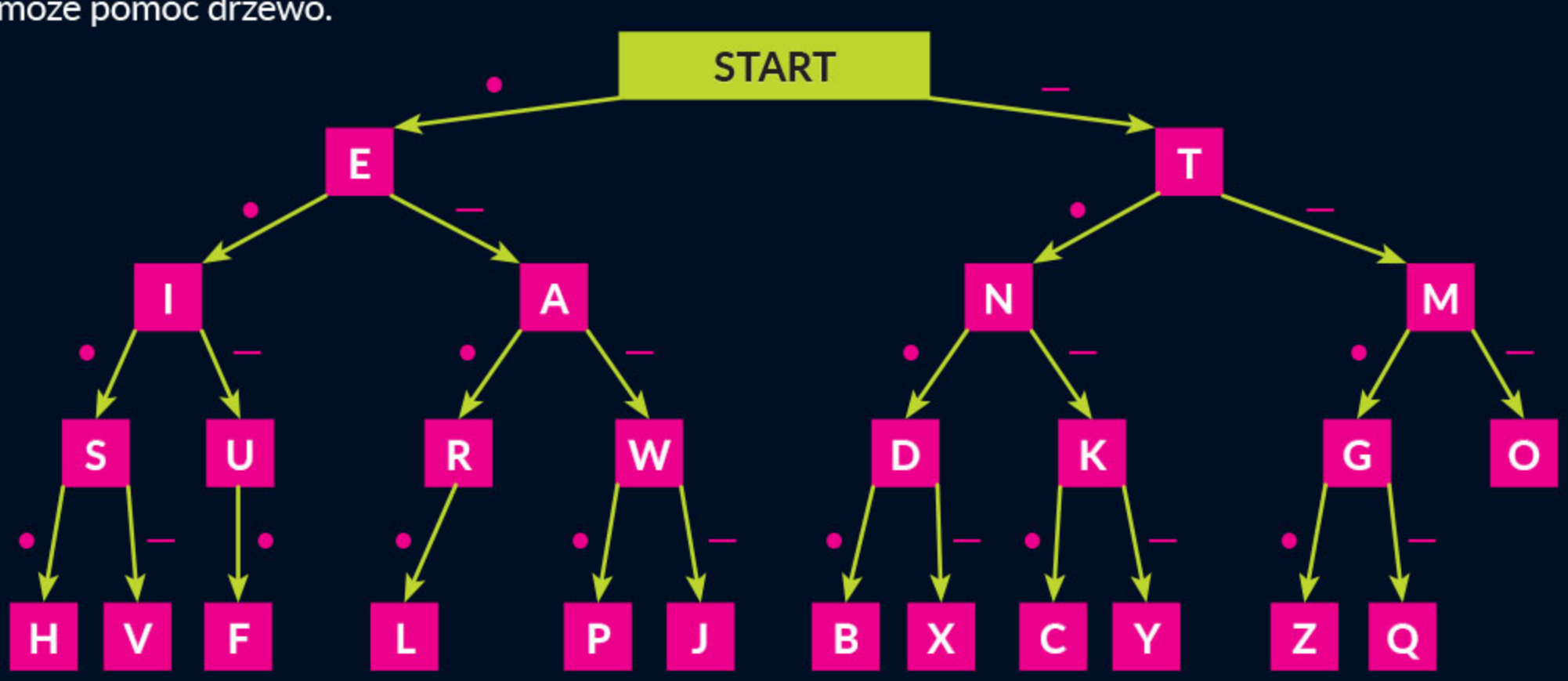
Litery, liczby i znaki specjalne w alfabecie opracowanym w 1832 r. przez amerykańskiego malarza Samuela Morse'a są przedstawiane jako kropki i kreski. Kod, który obecnie jest używany głównie przez radioamatorów, może służyć do szyfrowania tekstu pisanego oraz do sygnalizacji światłem, dźwiękiem czy chorągiewkami.

**PRZYKŁAD:** SOS ➡ /• • • /– – – /• • •/

| LITERA | KOD | LITERA | KOD |
|---|---|---|---|
| A | • – | N | – • |
| B | – • • • | O | – – – |
| C | – • – • | P | • – – • |
| D | – • • | Q | – – • – |
| E | • | R | • – • |
| F | • • – • | S | • • • |
| G | – – • | T | – |
| H | • • • • | U | • • – |
| I | • • | V | • • • – |
| J | • – – – | W | • – – |
| K | – • – | X | – • • – |
| L | • – • • | Y | – • – – |
| M | – – | Z | – – • • |



W odczytaniu wiadomości zapisanej za pomocą alfabetu Morse'a może pomóc drzewo.



**Szyfrowanie polega na przekształcaniu wiadomości, tzw. tekstu jawnego, w szyfrogram tak, aby uniemożliwić osobom niepowołanym odczytanie informacji.**

## ROT13

Alfabet o długości 26 znaków dzieli się na dwie części, a następnie każdej literze z pierwszej połowy przypisuje się odpowiadającą jej literę z drugiej połowy i na odwrót, np.: A → N, N → A, J → W, W → J. Ten sam algorytm wykorzystywany jest do szyfrowania i do deszyfrowania wiadomości. Jest to szczególny przypadek szyfru Cezara z kluczem 13.

**PRZYKŁAD:**
TAJNE HASLO BRZMI – CUKIEREK. ➞ GNWAR UNFYB OEMZV – PHXVRERX.

| A | B | C | D | E | F | G | H | I | J | K | L | M |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ↓↑ | ↓↑ | ↓↑ | ↓↑ | ↓↑ | ↓↑ | ↓↑ | ↓↑ | ↓↑ | ↓↑ | ↓↑ | ↓↑ | ↓↑ |
| N | O | P | Q | R | S | T | U | V | W | X | Y | Z |

## CZEKOLADKA

W tym szyfrze podpowiedzią okazuje się rysunek. Aby zaszyfrować literę, należy narysować „kostkę czekoladki" i postawić kropkę bliżej tej krawędzi, przy której jest szyfrowana litera. Aby zapisać litery T, U, Y lub W, rysuje się odpowiedni „dzióbek", do zapisania litery Z służy okrąg.



**PRZYKŁAD:**
NAMIOT ➞






### PRACA Z INFOGRAFIKĄ

1. Które z podanych szyfrów są podstawieniowe, a które przestawieniowe?
2. Wymień litery, które w alfabecie Morse'a są najkrótsze i najdłuższe. Dlaczego tak wybrano kod?



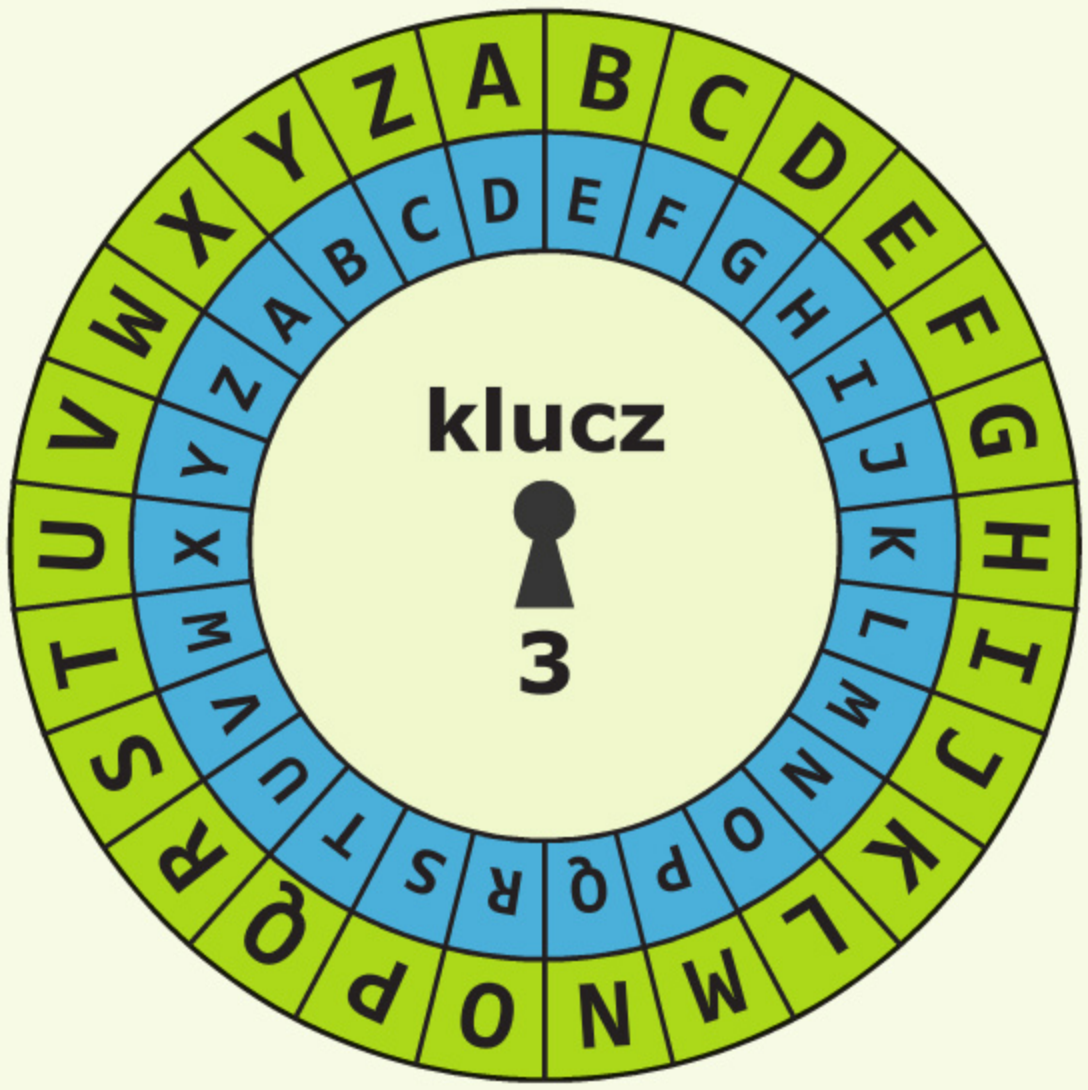
# 18. Szyfrowanie i deszyfrowanie tekstu

- Podstawowe pojęcia kryptograficzne
- Szyfr Cezara
- Szyfrowanie i odszyfrowywanie za pomocą kodów ASCII

## ZADANIE NA START

**Odczytaj wiadomość**

Wiesz, że wiadomość **spotkanie o czwartej** zaszyfrowano za pomocą poniższego koła do szyfrowania jako **vsrwndqlh r fczduwhm**. Na tej podstawie odszyfruj wiadomość **vhnuhwqb olvw srg vfkrgdpl**.




## SZYFR CEZARA

W szyfrze podstawieniowym stosowanym przez Juliusza Cezara każdy znak/każda litera tekstu jawnego były zastępowane innymi według przyjętego klucza, znanego zarówno odbiorcy, jak i nadawcy wiadomości – i wykorzystywanego do tworzenia szyfrogramu i jego odszyfrowywania.

**ZAPAMIĘTAJ!**

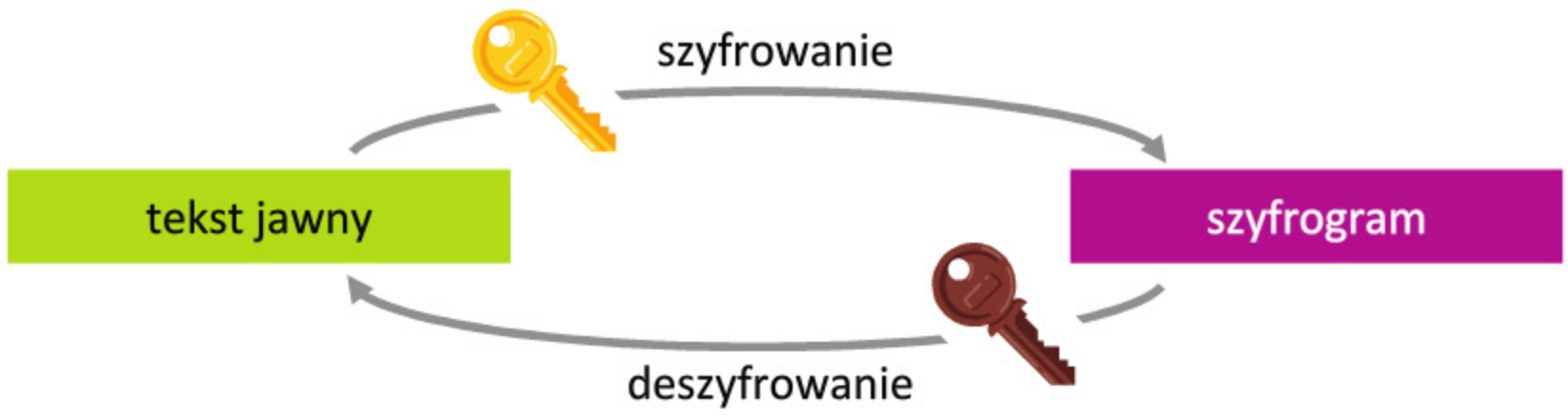
**Klucz** to informacja umożliwiająca wykonywanie pewnej czynności kryptograficznej, np. szyfrowania lub deszyfrowania.

Klucz w szyfrze Cezara to stała wartość, która wskazuje, o ile w prawo litera zaszyfrowana jest oddalona od pierwotnej. Dla danego napisu *t* i klucza *k* każdej literze napisu *t* przyporządkowuje się literę występującą o *k* miejsc dalej w alfabecie. Po dojściu do końca alfabetu należy zacząć liczyć od początku. Klucz 3 oznacza, że literze **a** odpowiada litera **d**, literze **b** – **e**, a literze **z** – **c**. Klucz 7 oznacza natomiast, że literze **a** odpowiada litera **h**, literze **b** – **i**, a literze **z** – **g**. Rzymski wódz i polityk szyfrował listy do swoich przyjaciół za pomocą klucza 3.

Gdy ktoś ma tekst jawny i klucz, może uzyskać szyfrogram. Jedna metoda szyfrowania pozwala na uzyskanie różnych szyfrogramów dla różnych kluczy.

**POSZUKAJ W INTERNECIE**

Sprawdź, na czym polega steganografia.




**Rys. 1.** Proces szyfrowania i deszyfrowania informacji

## WYKORZYSTANIE KODÓW ASCII

Do szyfrowania można użyć kodu ASCII (*American Standard Code for Information Interchange*), który przyporządkowuje każdemu znakowi w komputerze liczbę z zakresu 0–127 (np. kod ASCII litery **a** to 97, litery **k** – 107, litery **z** – 122).

Aby otrzymać kod ASCII danego znaku, należy wykorzystać funkcję **ord(znak)** – jeśli zastosuje się np. polecenie **ord("a")**, program wyświetli liczbę **97**, a jeśli poniższy skrypt – wyświetli litery i odpowiadające im kody ASCII od **97** do **103**.

```
1. for znak in "abcdefg":
2.     print(znak + ":", ord(znak))
```

**Rys. 2.** Skrypt pokazujący znaki podane w sekwencji pętli **for** oraz odpowiadające im kody ASCII

Działanie odwrotne wykonuje funkcja **chr(kod)**. Pozwala ona otrzymać znak odpowiadający danemu kodowi ASCII. Jeśli zastosuje się np. polecenie **chr(97)**, program wyświetli literę **a**, a jeśli skrypt poniżej – litery **z**, **y**, **x**, **w**, **v**, **u**, **t**.

```
1. for kod in range(122, 115, -1):
2.     print(chr(kod))
```

**Rys. 3.** Skrypt umożliwiający wypisanie znaków na podstawie kodów ASCII podanych jako parametry w pętli **for**

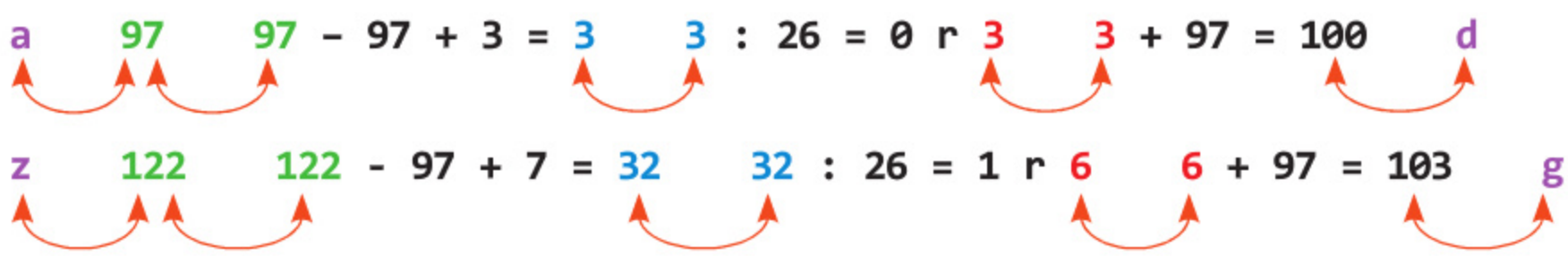
## Ćwiczenie 1. Szyfrowanie znaku

Przeanalizuj dane w tabeli i zdefiniuj funkcję **szyfruj_znak(znak, klucz)**, której parametrami są – odpowiednio – dowolna mała litera alfabetu łacińskiego oraz liczba całkowita nieujemna mniejsza od 1000, a wynikiem jest litera zaszyfrowana danym kluczem.

| Wywołanie funkcji | Wynik |
|---|---|
| `szyfruj_znak("a", 3)` | `"d"` |
| `szyfruj_znak("z", 7)` | `"g"` |

1. Przeanalizuj treść ćwiczenia i podane przykłady.

- Aby otrzymać pozycję danej litery w alfabecie, trzeba zamienić literę na kod ASCII, a następnie odjąć od niej kod ASCII litery a (**97**). Do liczby w ten sposób otrzymanej należy dodać klucz. Dla uzyskanej sumy trzeba wyznaczyć resztę z dzielenia jej przez **26** (długość alfabetu łacińskiego). Na koniec do wyniku wystarczy dodać kod ASCII litery a (**97**) i zamienić tak uzyskany kod ASCII na literę.




2. Zapisz odpowiednią formułę.

```
chr((ord(znak) - 97 + klucz) % 26 + 97)
```

3. Zdefiniuj funkcję **szyfruj_znak(znak, klucz)**.

```
1. def szyfruj_znak(znak, klucz):
2.     return chr((ord(znak) - 97 + klucz) % 26 + 97)
```

4. Przetestuj działanie skryptu z różnymi danymi. Jaki klucz trzeba podać, aby zaszyfrować tekst tym samym znakiem, który pojawia się w tekście jawnym?

## Ćwiczenie 2. Szyfrowanie tekstu

Przeanalizuj dane w tabeli i zdefiniuj funkcję **szyfruj(tekst, klucz)**, której parametrami są – odpowiednio – ciąg małych liter alfabetu łacińskiego oraz liczba całkowita nieujemna mniejsza od 1000, a wynikiem jest tekst zaszyfrowany danym kluczem.

| Wywołanie funkcji | Wynik |
|---|---|
| `szyfruj("kiedyspotkanie", 13)` | `"xvrqlfcbgxnavr"` |
| `szyfruj("tajne", 133)` | `"wdmqh"` |

1. Przeanalizuj treść ćwiczenia i podane przykłady.

2. Opracuj algorytm.

- Utwórz zmienną pomocniczą **pom**, która będzie stanowić napis pusty, a następnie dopisz do niej każdą kolejną zaszyfrowaną literę tekstu, np. dla **kiedyspotkanie** będą to kolejno litery **x**, **v**, **r**, ..., **a**, **v**, **r**. Oznacza to, że wartości zmiennej **pom** to: **"x"**, **"xv"**, "**xvr**", ..., **"xvrqlfcbgxnavr"**. W ten sposób wynikiem funkcji będzie ostatnia wartość zmiennej **pom**.

❸ Zapisz funkcję **szyfruj(tekst, klucz)**.

```
1. def szyfruj(tekst, klucz):
2.     pom = ""
3.     for znak in tekst:
4.         pom = pom + szyfruj_znak(znak, klucz)
5.     return pom
```

❹ Sprawdź działanie skryptu z różnymi danymi. Zastanów się, jak za pomocą tej funkcji odszyfrować tekst.

## Ćwiczenie 3. Odszyfrowywanie tekstu

Wynikiem zdefiniowanej poniżej funkcji **deszyfruj(tekst, klucz)** powinien być odszyfrowany tekst z danym kluczem.

```
1. def deszyfruj(tekst, klucz):
2.     return szyfruj(tekst, 26 + klucz)
```

Przeanalizuj dane w tabeli i popraw błąd w skrypcie.

| Wywołanie funkcji | Wynik |
|---|---|
| **deszyfruj("xvrqlfcbgxnavr", 13)** | **"kiedyspotkanie"** |
| **deszyfruj("wdmqh", 3)** | **"tajne"** |

❶ Przeanalizuj treść ćwiczenia i podane przykłady.

- Odszyfrowanie tekstu to w istocie ponowne zaszyfrowanie z kluczem będącym jego dopełnieniem do 26. Jeśli tekst jawny zaszyfrowano z kluczem 13, to odszyfrowanie polega na zaszyfrowaniu kryptogramu kluczem 26 – 13 = 13 (co oznacza, że należy przesunąć się do przodu o 13 liter).

**WIEM WIĘCEJ**

**Szyfr Cezara** stosunkowo łatwo złamać, czyli odczytać wiadomość bez znajomości klucza. Chociaż wartość klucza jest nieograniczona, to praktycznie istnieje tylko **26 możliwości**. Automatyczna technika łamania polega na analizie częstości liter. Jeżeli tekst jest dostatecznie długi i wiadomo, w jakim napisano go języku, można zbadać częstość występowania liter w szyfrogramie i porównać ją z częstością występowania liter w danym języku. Najlepsze dopasowanie z dużym prawdopodobieństwem przełoży się na klucz.

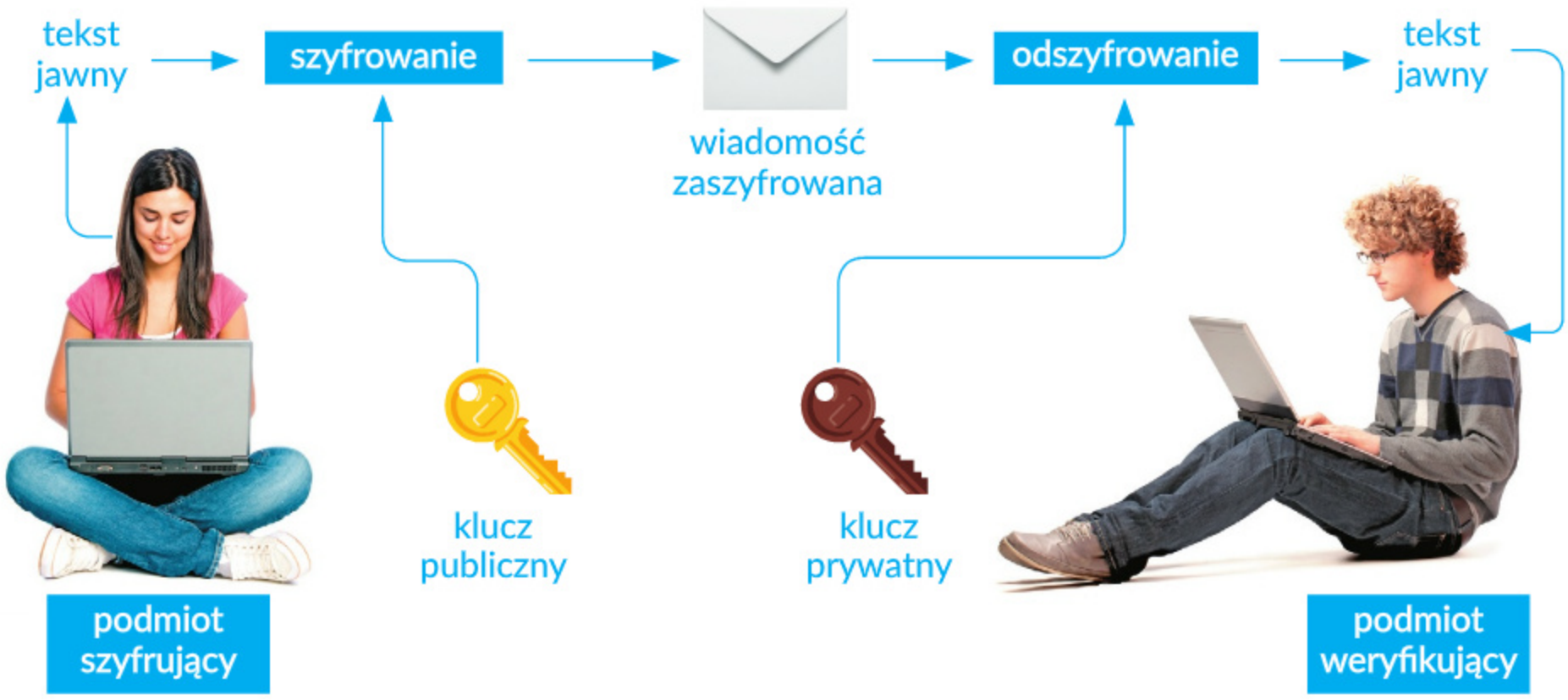
### Ćwiczenie 4. Inaczej na pozycjach nieparzystych i parzystych

Przeanalizuj dane w tabeli i zdefiniuj funkcję **`szyfruj2(tekst, klucz1, klucz2)`**, której parametrami są – odpowiednio – ciąg małych liter alfabetu łacińskiego oraz dwie liczby całkowite, a wynikiem jest tekst zaszyfrowany w ten sposób, że litery na pozycjach parzystych zaszyfrowano kluczem **`klucz1`**, a na nieparzystych – kluczem **`klucz2`**.

| Wywołanie funkcji | Wynik |
|---|---|
| `szyfruj2("poufnytekst", 3, 10)` | `"zreixbdhuvd"` |
| `szyfruj2("poczekajnamnie", 123, 5)` | `"uhhsjdfcstrgnx"` |

## ZASTOSOWANIE SZYFROWANIA

Dzięki wykorzystaniu komputera szyfrowanie i deszyfrowanie przebiega szybciej, niż gdyby szyfrowało się informację samodzielnie. Łatwiej jest jednak złamać szyfr komputerowy, chociażby dlatego, że można wówczas wielokrotnie wykonać algorytm odszyfrowywania. W praktyce stosuje się więc bardziej skomplikowane **metody szyfrowania, oparte na kryptografii asymetrycznej**. Zakłada ona istnienie dwóch kluczy: prywatnego, który zna tylko jego właściciel, i publicznego (jawnego), przy czym klucza prywatnego nie da się łatwo odtworzyć na podstawie publicznego. Szyfry asymetryczne wykorzystuje się m.in. w celu zabezpieczenia dostępu do zasobów hasłem lub cyfrowego podpisywania dokumentów.




**Rys. 4.** Schemat szyfrowania opartego na kryptografii asymetrycznej

## PODSUMOWANIE

- W szyfrze Cezara do szyfrowania i deszyfrowania wiadomości stosuje się klucz, który wskazuje, o ile pozycji w alfabecie należy przesunąć literę.

- Klucz to informacja umożliwiająca wykonywanie pewnej czynności kryptograficznej, np. szyfrowania lub deszyfrowania. Gdy ktoś ma tekst jawny i klucz, może uzyskać szyfrogram. Jedna metoda szyfrowania pozwala na uzyskanie różnych szyfrogramów dla różnych kluczy.
- Do szyfrowania tekstów złożonych z liter alfabetu łacińskiego warto wykorzystywać kody ASCII.
- Obecnie w kryptografii stosuje się szyfry asymetryczne oparte na dwóch kluczach – prywatnym i publicznym.

## PYTANIA SPRAWDZAJĄCE

1. Ile różnych kluczy jest w szyfrze Cezara?
2. Jakie wartości kluczy powodują, że tekst jawny i szyfrogram są identyczne?
3. Gdzie obecnie stosuje się elementy kryptografii?

## PYTANIA QUIZOWE

1. Jak nazywa się szyfr, w którym każdą literę tekstu jawnego zastępuje się inną?
   **a.** Podstawieniowy.
   **b.** Przestawieniowy.
   **c.** Symetryczny.
   **d.** Asymetryczny.
2. Ile wynosi kod ASCII litery `m`, jeśli kod ASCII litery `k` wynosi 107?
   **a.** 108
   **b.** 109
   **c.** 110
   **d.** 111
3. Napis **`tajemnica`** zaszyfrowano szyfrem Cezara i otrzymano **`xeniqrmge`**. Jakiego klucza użyto?
   **a.** 1
   **b.** 2
   **c.** 3
   **d.** 4

## ZADANIA DODATKOWE

### Zadanie 1. Jakim kluczem zaszyfrowano?

Przeanalizuj dane w tabeli i zdefiniuj funkcję **`jaki(tekst_jawny, szyfrogram)`**, której parametrami są ciągi małych liter alfabetu łacińskiego, a wynikiem jest klucz z przedziału **`[0, 25]`**, pozwalający zaszyfrować tekst jawny za pomocą metody Cezara,

lub wartość **–1**, gdy szyfrogram nie mógł powstać z tekstu jawnego w wyniku szyfrowania metodą Cezara.

| Wywołanie funkcji | Wynik |
|---|---|
| `jaki("niespodzianka", "pkgurqfbkcpmc")` | `2` |
| `jaki("zmiana", "anjbxb")` | `-1` |

### Zadanie 2. Szyfrowanie polskiego tekstu

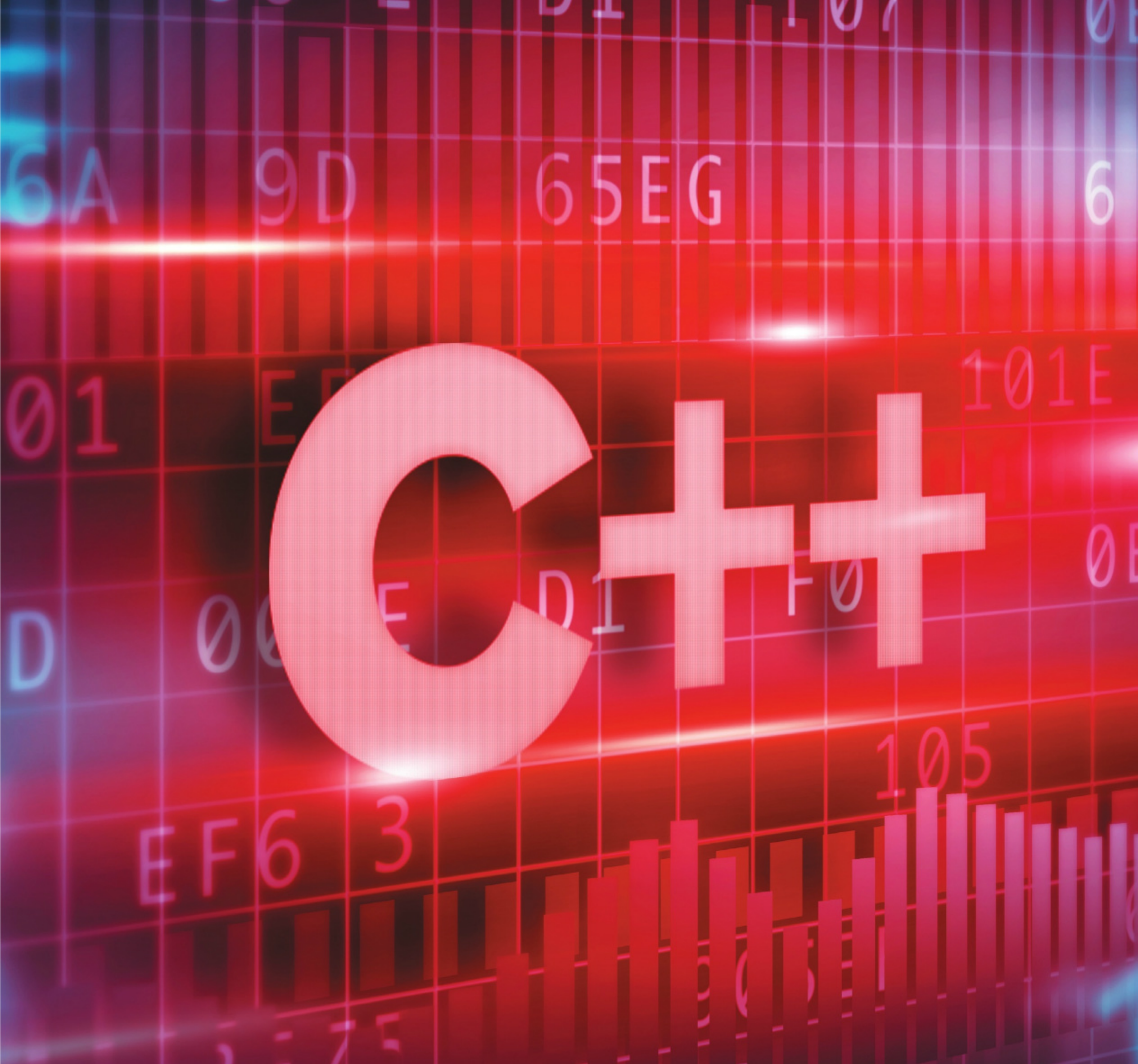
Przeanalizuj dane w tabeli i zdefiniuj funkcję **`szyfrujpl(tekst, klucz)`**, której parametrami są – odpowiednio – ciąg małych liter alfabetu (mogą w nim występować litery typowo polskie) oraz liczba całkowita nieujemna mniejsza od 1000, a wynikiem jest zaszyfrowany tekst z danym kluczem.

| Wywołanie funkcji | Wynik |
|---|---|
| `szyfrujpl("przekaż", 7)` | `"yzćkóęe"` |
| `szyfrujpl("wiadomość", 3)` | `"źlcfrorwę"` |

**Wskazówka:** zdefiniuj zmienną **`alfabet`**, będącą napisem złożonym ze wszystkich liter alfabetu polskiego, przy czym pozycja litery w napisie odpowiada kodowi litery.

### Zadanie 3. Złożenie szyfrów

Zaproponuj metodę szyfrowania opartą na złożeniu dwóch szyfrów, czyli na zastosowaniu jednego szyfru, a potem drugiego – do otrzymanego szyfrogramu. Napisz funkcję szyfrującą i funkcję deszyfrującą.




# ALGORYTMIKA I PROGRAMOWANIE W C++

Programowanie, które łączy myślenie kreatywne, abstrakcyjne, logiczne i analityczne, pozwala na rozwijanie kompetencji najbardziej pożądanych przez pracodawców, czyli:

- umiejętność rozwiązywania problemów krok po kroku za pomocą dzielenia problemów na mniejsze podproblemy,
- zdolność do uogólniania, testowania i oceny efektywności rozwiązania.

# PODSTAWOWE KONSTRUKCJE PROGRAMISTYCZNE

## SEKWENCJA POLECEŃ

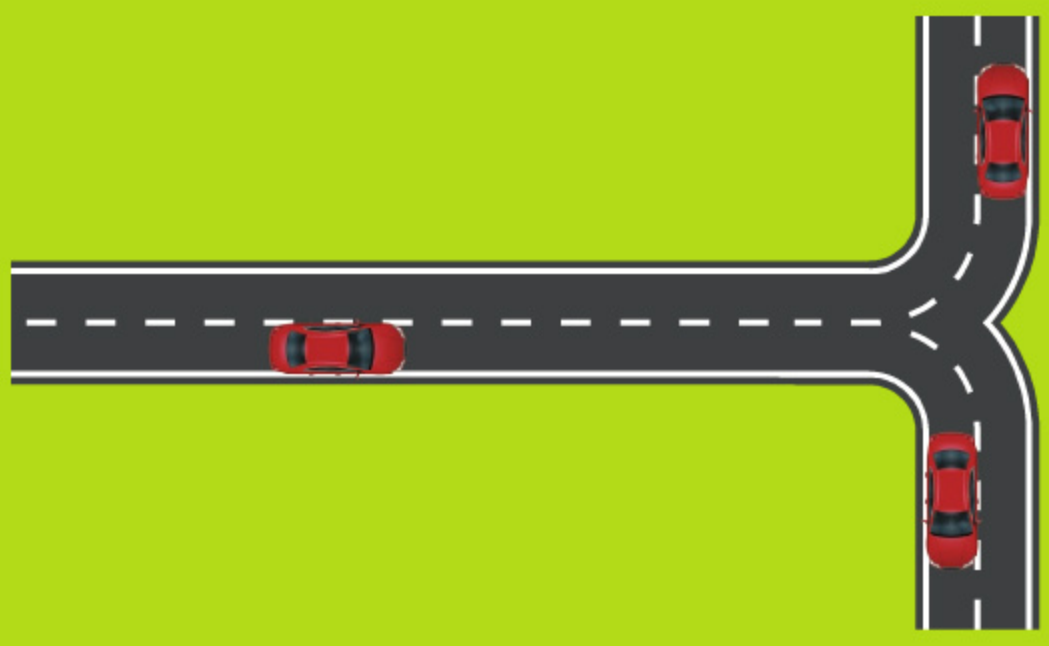
Polecenia są wykonywane jedno po drugim.



Polecenie dla kierowcy: jedź prosto.

## INSTRUKCJA WARUNKOWA

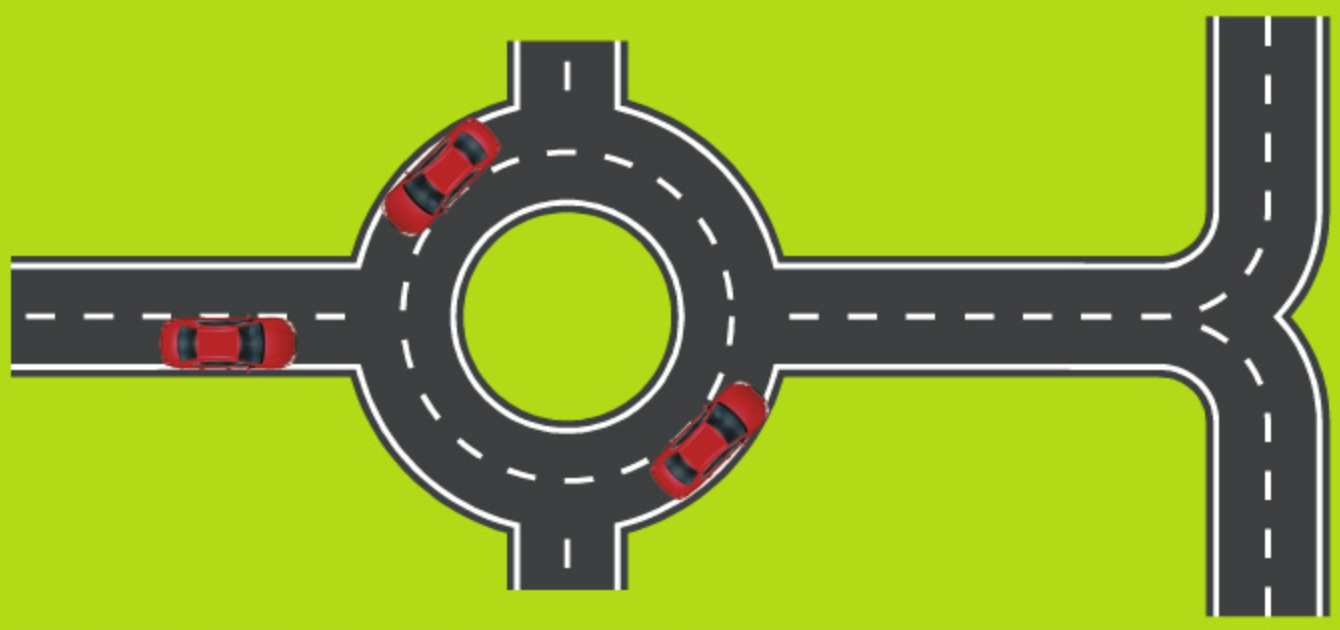
Pozwala na wykonanie różnych instrukcji w zależności od spełnienia określonego warunku.



Polecenie dla kierowcy: jeśli chcesz dojechać do szkoły, skręć w lewo; jeśli chcesz dotrzeć do sklepu, skręć w prawo.

## PĘTLA FOR

Powtarza zawarte w niej polecenia konkretną liczbę razy.



Polecenie dla kierowcy: na rondzie wykonaj trzy okrążenia.

## PĘTLA WHILE

Powtarza zawarte w niej polecenia, dopóki zachodzi podany warunek.

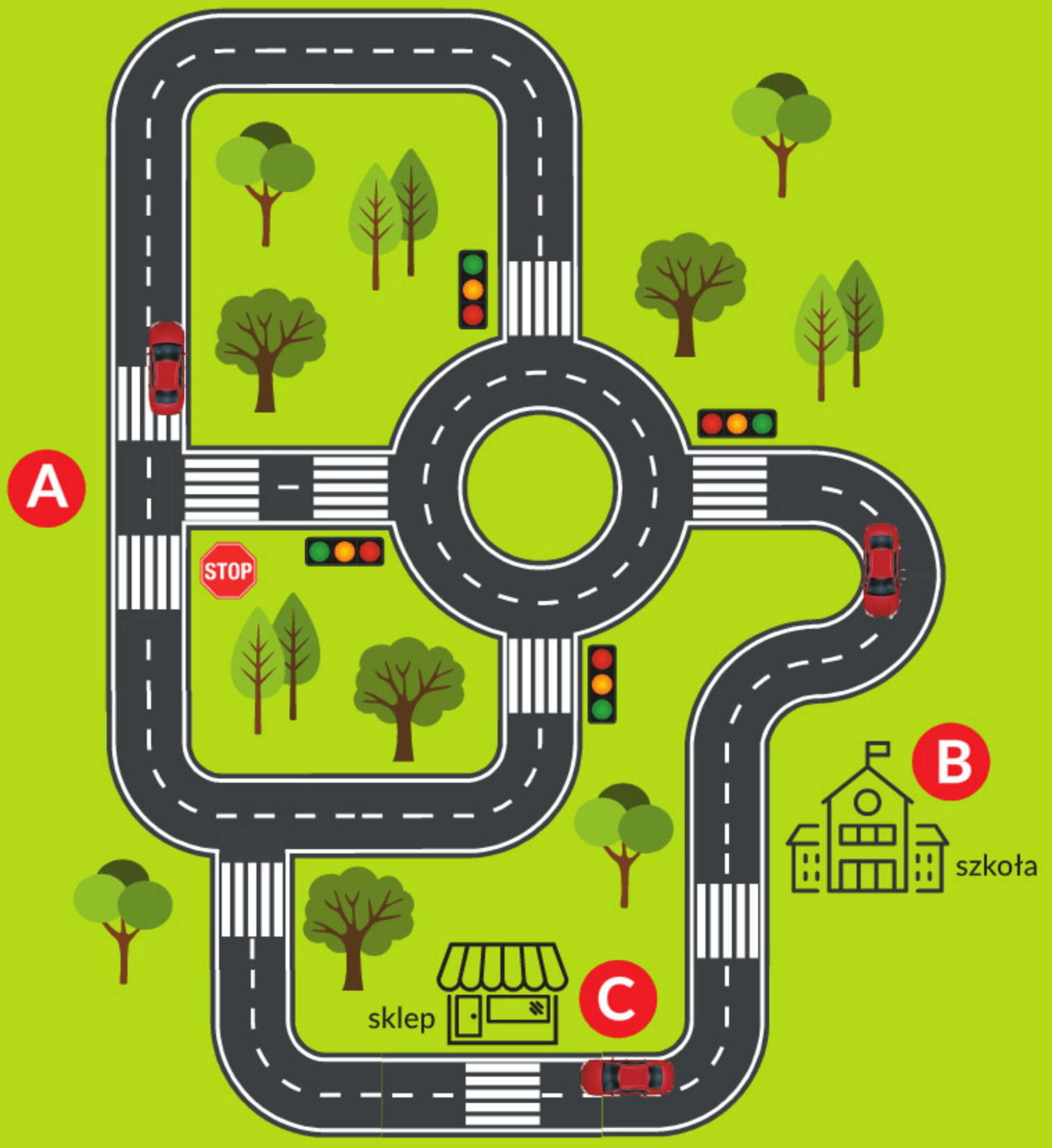


Polecenie dla kierowcy: na rondzie wykonuj okrążenia dopóty, dopóki zjazd nie zostanie odblokowany przez policjanta.

**Każdy język programowania ma swoją specyfikę, jednak pewne konstrukcje są wspólne dla wszystkich powszechnie używanych języków. Sprawdź, czy znasz podstawy, zanim zaczniesz uczyć się konkretnego języka.**

## FUNKCJA

Grupuje polecenia, które tworzą określoną logiczną całość i po wywołaniu funkcji są wykonywane w określonej kolejności.




Polecenie dla kierowcy: aby dojechać do sklepu (punkt C), jedź w kierunku szkoły (punkt B), a potem tak, jak prowadzi droga.



### PRACA Z INFOGRAFIKĄ

1. Które instrukcje powodują rozgałęzienie programu (różne ścieżki), a które nie?
2. Czym różnią się pętle `for` i `while`?



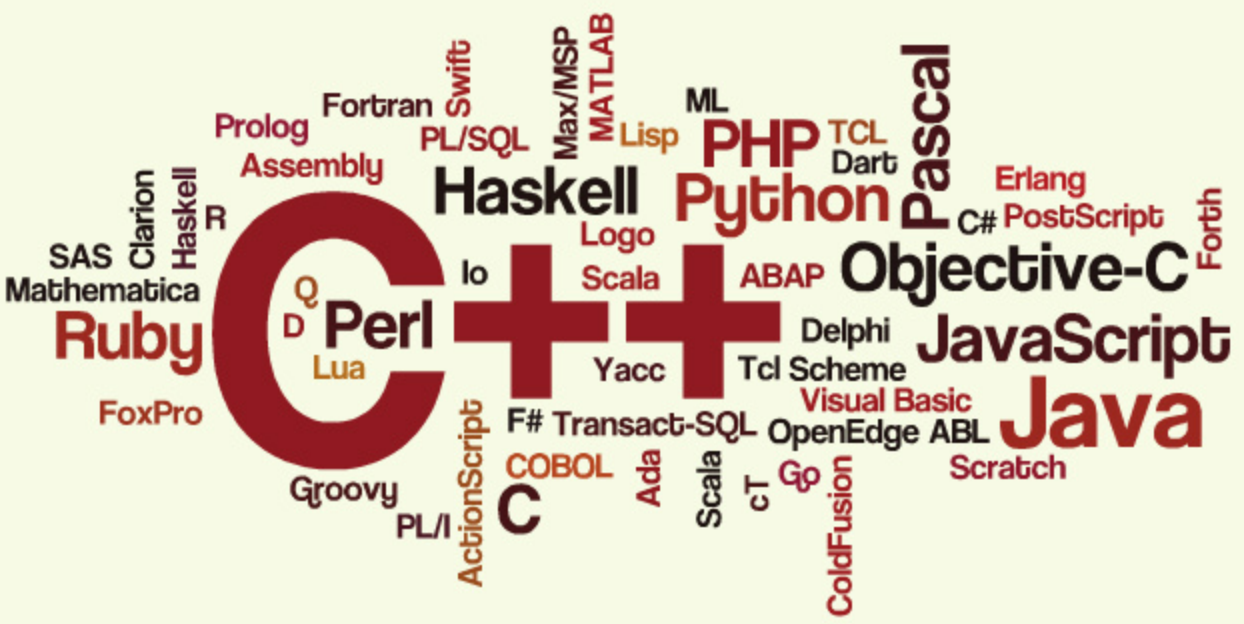
# 14A. Podstawy pracy w środowisku C++

- Wprowadzenie do programowania w języku C++
- Wzorzec programu
- Definiowanie prostych funkcji

## ZADANIE NA START

### Programowanie w C++

C++ to jeden z najbardziej znanych języków na świecie. Można w nim pisać aplikacje, które powinny szybko działać. Dlatego język ten jest wykorzystywany m.in. do tworzenia systemów operacyjnych. Znajdź trzy aplikacje napisane w języku C++.



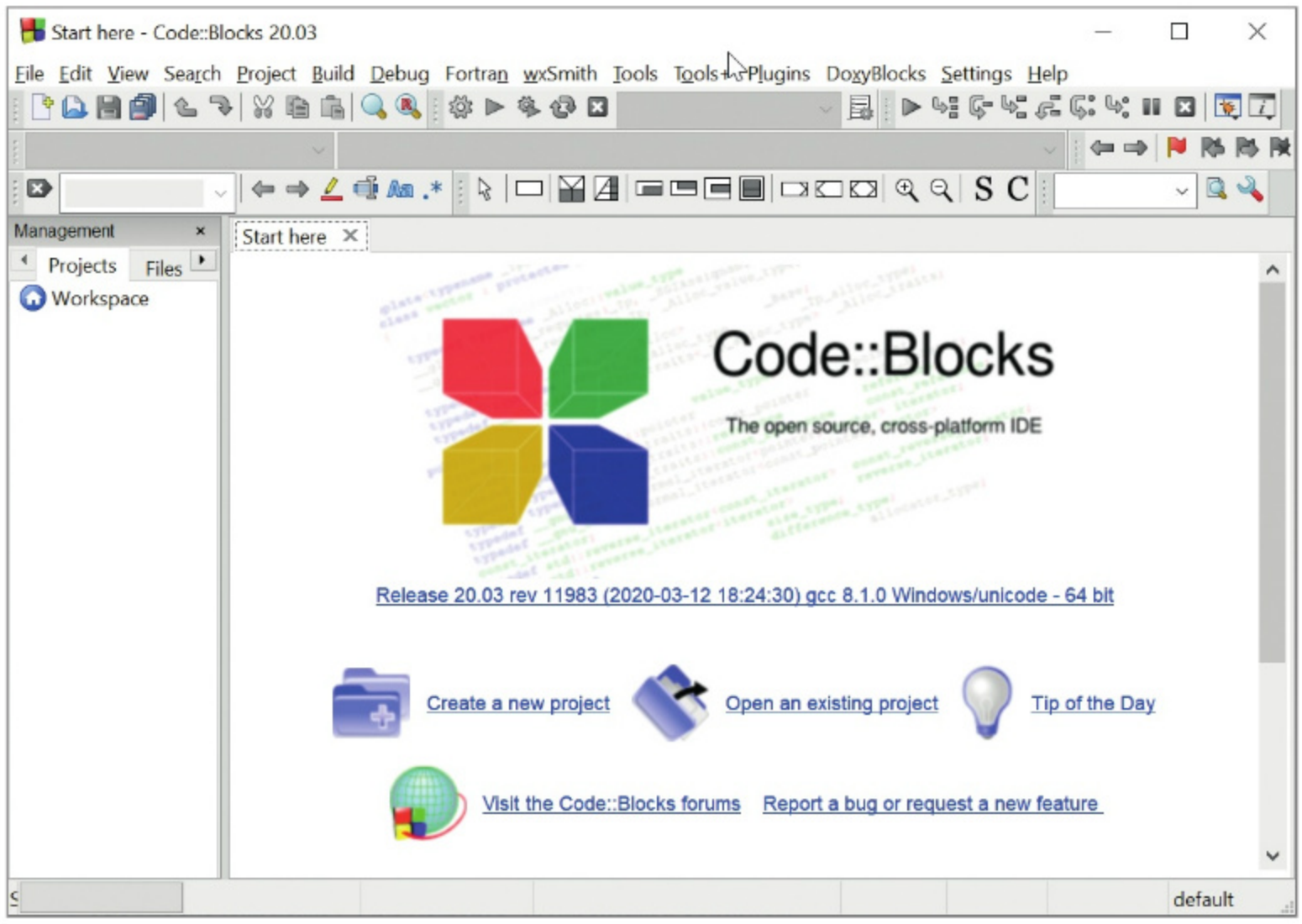

## ABC JĘZYKA C++

C++ to język wieloparadygmatowy, co oznacza, że można w nim jednocześnie stosować różne style programowania, np. dzielenie kodu na procedury wykonujące określone operacje, definiowanie programów za pomocą obiektów łączących dane i metody czy programowanie uogólnione, bez znajomości typów danych, na których kod będzie pracował.

**POSZUKAJ W INTERNECIE**

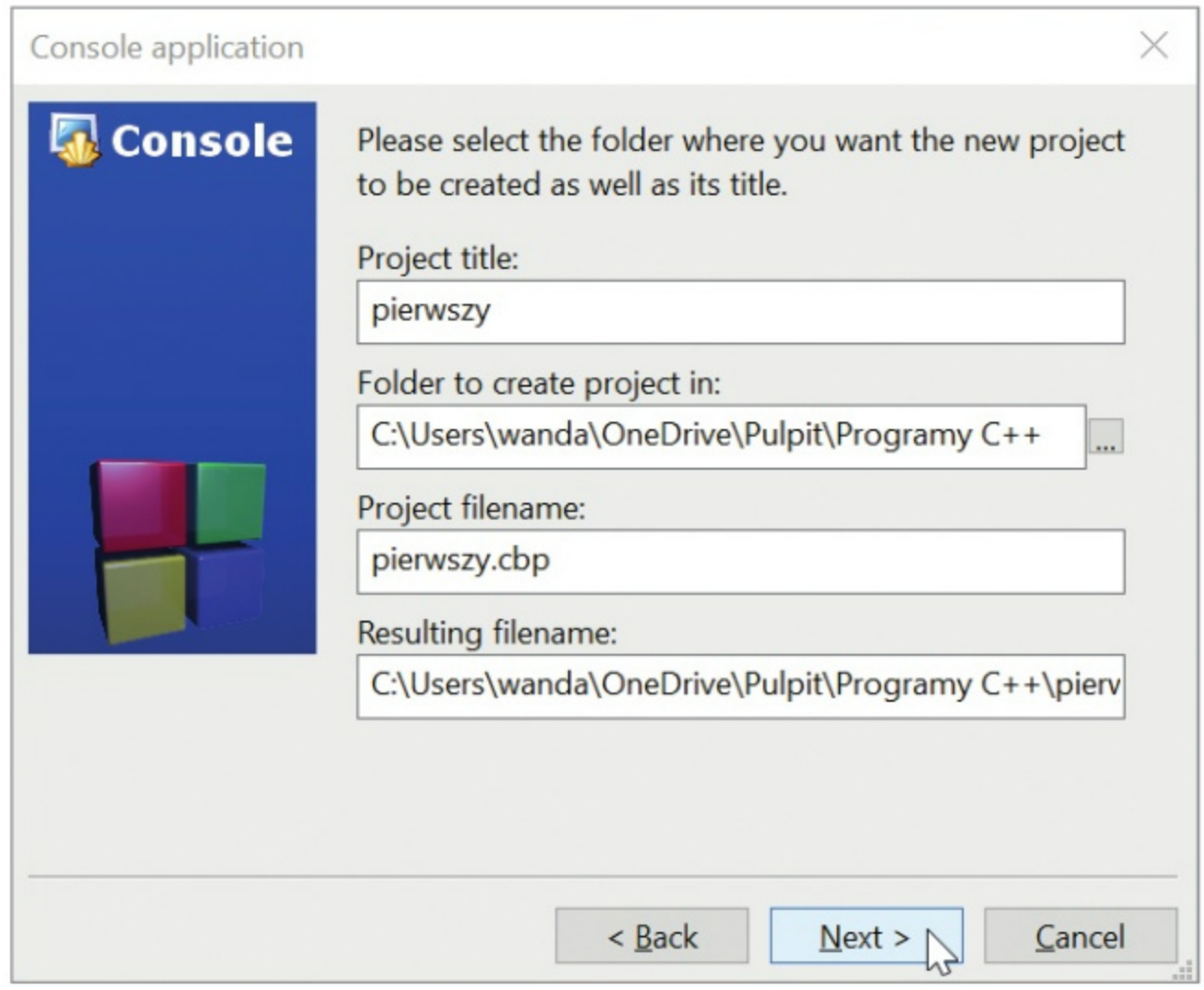
Poszukaj w internecie wywiadów z twórcą języka C++ Bjarnem Stroustrupem, aby dowiedzieć się więcej o nim i jego filozofii.

Kod programu napisanego w języku C++ zapisuje się jako plik tekstowy. Zwyczajowo podczas zapisywania pliku nadaje mu się rozszerzenie CPP i za pomocą kompilatora tworzy plik wykonywalny. Do pisania programów wygodnie jest używać zintegrowanego środowiska deweloperskiego (IDE), zawierającego różne funkcje. W systemach Windows, Linux lub OS pracę w środowisku C++ można rozpocząć od zainstalowania na komputerze środowiska Code::Blocks, które należy pobrać z serwisu **www.codeblocks.org**.



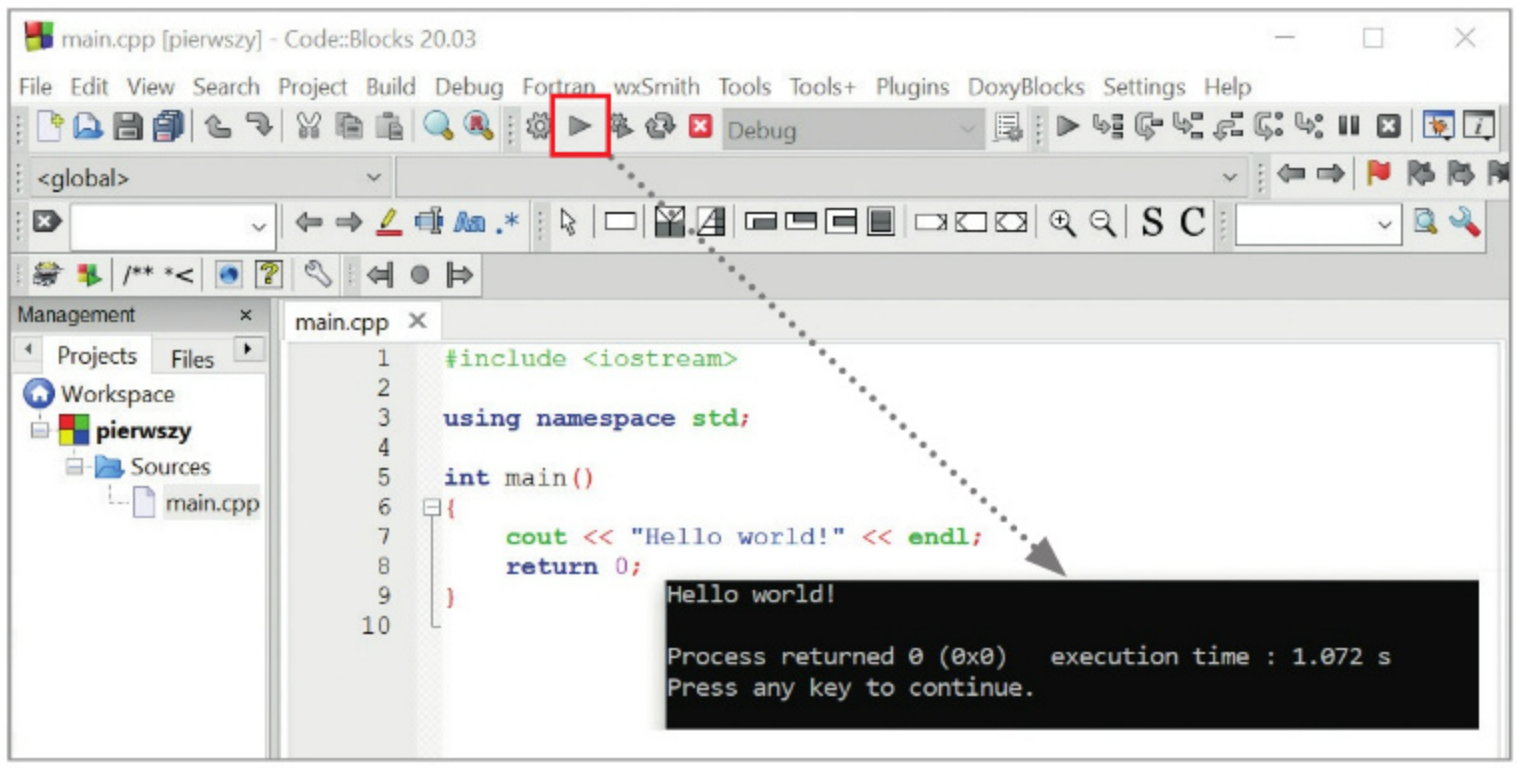

**Rys. 1.** Zintegrowane środowisko programistyczne Code::Blocks

Aby rozpocząć pracę w edytorze Code::Blocks, należy kliknąć **Create a new project** w oknie programu i wybrać **Console application**. Następnie, po wybraniu języka (C lub C++), należy wskazać miejsce zapisu projektu, wpisać jego tytuł i nazwę pliku.




**Rys. 2.** Tworzenie nowego projektu

Następnie w panelu **Management**, w zakładce **Projects** należy otworzyć folder **Sources** i dwukrotnie kliknąć plik **main.cpp**. Wyświetli się wzorzec programu z domyślnym programem wypisującym tekst „Hello world!”. Po zapisaniu kodu program uruchamia się za pomocą przycisku ▶ (**Run**).

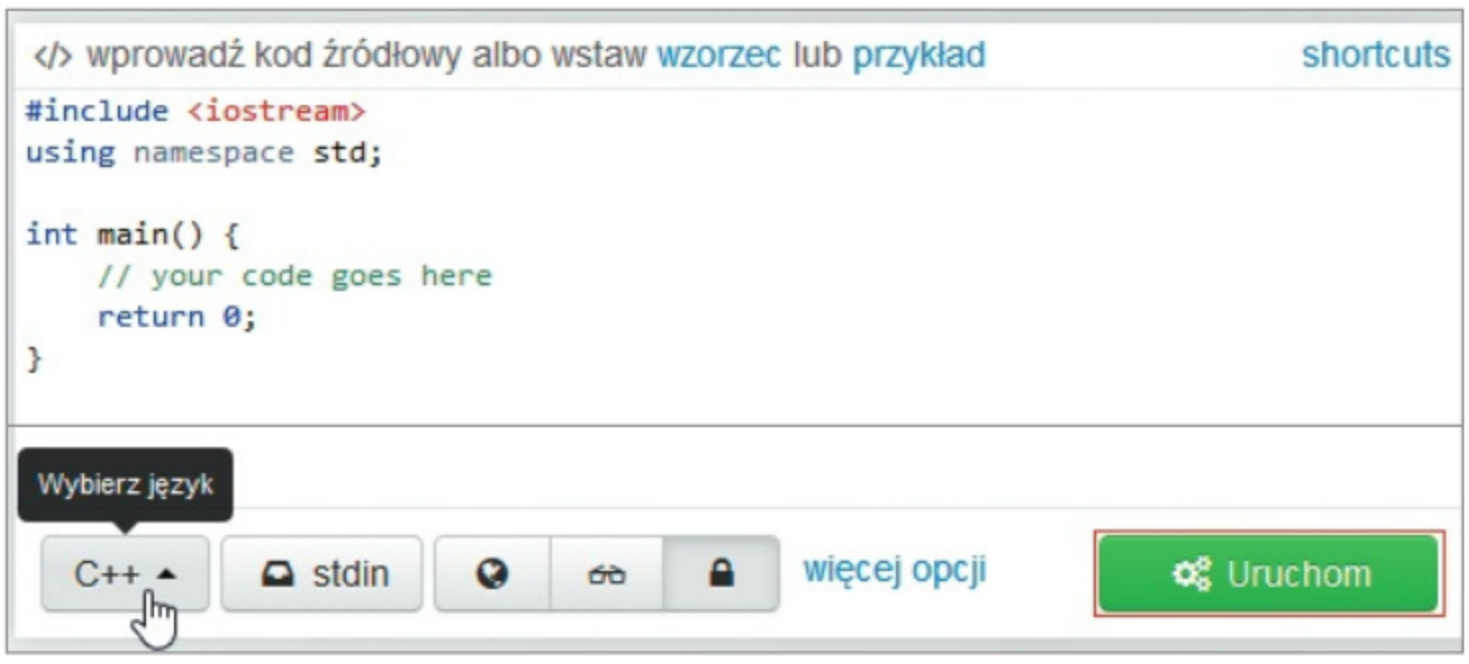



**Rys. 3.** Uruchamianie domyślnego programu w Code:Blocks

## ZAMIAST IDLE

Programy C++ można zapisywać w dowolnym edytorze tekstu, np. w zwykłym systemowym Notatniku, można też korzystać z coraz popularniejszych w ostatnich latach kompilatorów online, które obsługują wiele języków programowania (niektóre – nawet ponad 50).

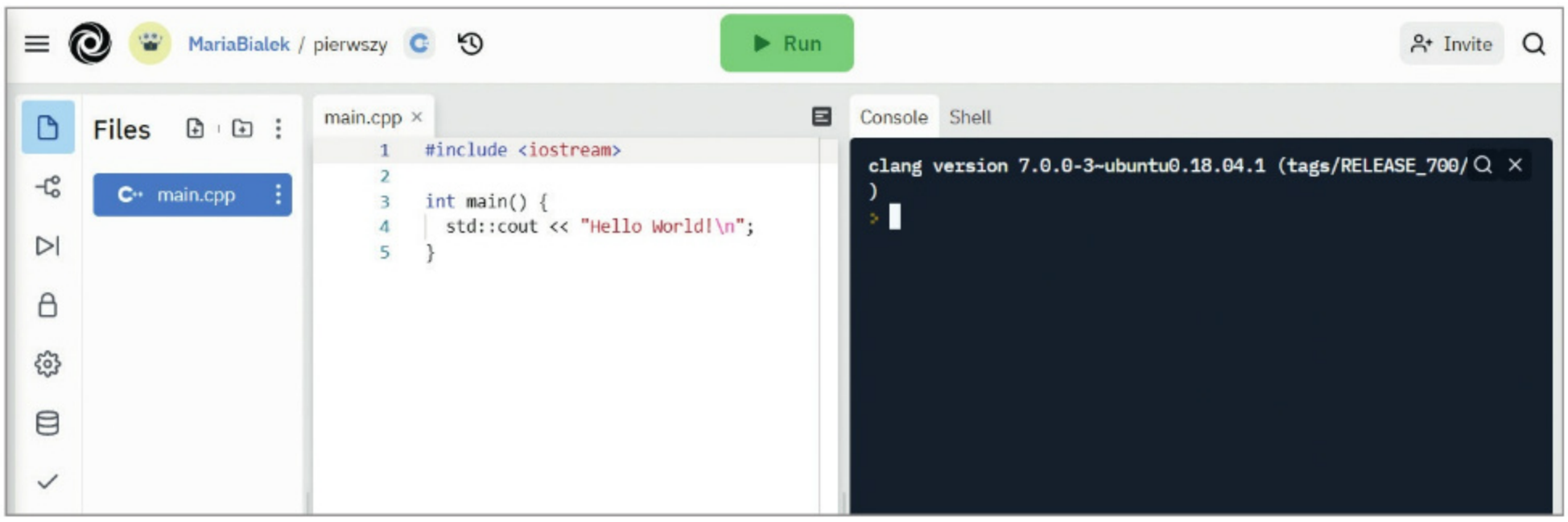
Pracę w chmurze umożliwia m.in. serwis **ideone.com,** stworzony przez polską firmę z Trójmiasta. Wystarczy w przeglądarce wpisać adres serwisu, po wgraniu strony wybrać z rozwijanej listy odpowiedni język i można zacząć wpisywać kod. Aby uruchomić program, należy kliknąć przycisk **Uruchom**. Warto założyć konto i ustawić domyślny język programowania, język interfejsu oraz włączyć podświetlanie składni, a także używać notatek do opisu programu oraz etykiet do organizacji swoich kodów. Wszystkie skrypty programów są zapisywane w chmurze i znajdują się w sekcji **moje kody**. Można je pobrać na dysk i skorzystać z nich ponownie.




**Rys. 4.** W serwisie **ideone.com** należy najpierw wybrać język. Aby uruchomić program, należy kliknąć przycisk **Uruchom**

Osoby swobodnie posługujące się językiem angielskim mogą skorzystać z serwisu programistycznego **replit.com**. W tym przypadku konieczne jest założenie konta lub zalogowanie się poprzez konto Google. Aby rozpocząć pracę, należy kliknąć przycisk

+ **Create Repl**, a następnie wybrać preferowany język, zatytułować projekt lub zaakceptować nazwę domyślną i zmiany wprowadzone w oknie zatwierdzić przyciskiem + **Create Repl**. Kod wpisuje się w środkowym polu. Aby uruchomić program, należy kliknąć przycisk **Run**. Efekt działania pojawi się w oknie z prawej strony.




**Rys. 5.** Tworzenie nowego projektu w serwisie **replit.com**

## WZORZEC PROGRAMU

Wzorzec programu zawiera: odwołanie do biblioteki **<iostream>**, służącej do obsługi strumieni (dzięki temu można wyświetlać dane na standardowym wyjściu – ekranie), określenie przestrzeni nazw **std** oraz funkcję główną **main()**. Nazwa funkcji jest zawsze poprzedzona określeniem typu wyniku – **int** albo **void** (**int** oznacza liczbę całkowitą, **void** informuje, że funkcja nie zwraca żadnej wartości). Treść funkcji umieszcza się w nawiasach klamrowych. Polecenie **return 0** oznacza, że działanie funkcji zakończyło się bez błędu.

```
1. #include <iostream>
2. using namespace std;
3.
4. int main() {
5.     // tu pisze się kod programu
6.     return 0;
7. }
```

Każde polecenie w C++ kończy się średnikiem.

W C++ jednowierszowy komentarz rozpoczyna się od dwóch ukośników //. Komentarze wielowierszowe są zawarte między znakami /* i */.

**Rys. 6.** Wzorzec programu

**W PRAKTYCE**

Oprócz bloków oznaczonych nawiasami klamrowymi warto dodatkowo stosować wcięcia – kod jest wtedy bardziej przejrzysty i zrozumiały.

## POLECENIE COUT

Programowanie polega na pisaniu instrukcji zgodnie z obowiązującymi regułami. W dalszej kolejności kod jest kompilowany, czyli tłumaczony na język zrozumiały przez komputer, a następnie wykonywany.

Instrukcja **cout** (czytaj: si-aut) wyświetla to, co zostanie napisane po symbolu **<<**, np. tekst (ciąg znaków), liczbę albo wartość zmiennej.

**ZAPAMIĘTAJ!**

```
cout << "<ciąg znaków>";
cout << <liczba>;
cout << <zmienna>;
```

Aby dodać pusty wiersz, można skorzystać z polecenia **cout << endl;**

Jeśli w zapisie pojawi się błąd, podczas kompilacji wyświetli się odpowiedni komunikat.

Brak średnika na końcu wiersza 7.

```
main.cpp
 1  #include <iostream>
 2
 3  using namespace std;
 4
 5  int main()
 6  {
 7      int a = 3
 8      cout << a + 1 << endl;
 9      return 0;
10  }

Logs & others
Code::Blocks | Search results | Cccc | Build log
File          L...  Message
                    === Build: Debug in ppp (compiler: GNU GCC C
C:\Users...         In function 'int main()':
C:\Users...   8     error: expected ',' or ';' before 'cout'
```

edytuj klonuj pobierz

```
1.  #include <iostream>
2.  using namespace std;
3.
4.  int main() {
5.      cout << 3 + 4 < endl;
6.      return 0;
7.  }
```

Błąd kompilacji #stdin błąd kompilacji #stdout 0s 0KB

Brak jednego znaku < podczas wypisywania.

stdin

```
Standard input is empty
```

informacja o kompilacji

```
prog.cpp: In function ‘int main()’:
prog.cpp:5:16: error: no match for ‘operator<’
ed overloaded function type>’)
  cout << 3 + 4 < endl;
  ~~~~~~~~~~~~~~~^~~~~~
```

Próba wypisania wartości niezadeklarowanej zmiennej **ab**.

```
1  #include <iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      int abc = 2;
8      cout << ab << endl;
9      return 0;
10 }
```

```
> clang++-7 -pthread -std=c++17 -o main main.cpp
main.cpp:8:13: error: use of undeclared identifier 'ab'
    cout << ab << endl;
            ^
1 error generated.
exit status 1
>
```

**Rys. 7.** Przykładowe komunikaty błędów w edytorze Code::Blocks oraz w serwisach **ideone.com** i **replit.com**

## Ćwiczenie 1. Wypisywanie tekstu

Wypisz na ekranie trzy wybrane przez siebie hasła na temat podróżowania. Każde z nich powinno wyświetlać się w osobnym wierszu.

1. W edytorze kodu źródłowego w pierwszym wierszu funkcji głównej **`main()`** wpisz polecenie **`cout`** i wstaw symbol **`<<`**.
2. Wpisz hasło. Pamiętaj o cudzysłowach.
3. Zakończ wiersz symbolem **`<<`**, słowem kluczowym **`endl`** i średnikiem.
   - Ponieważ po ostatnim haśle nie musisz przechodzić do nowego wiersza, możesz pominąć **`<< endl`**.

```
1. #include <iostream>
2. using namespace std;
3.
4. int main() {
5.     cout << "Zwiedzaj świat." << endl;
6.     cout << "Należy podróżować, by się czegoś nauczyć." << endl;
7.     cout << "Raz do roku pojedź w miejsce, w którym jeszcze nie byłeś.";
8.     return 0;
9. }
```

4. Skompiluj i uruchom program.

### OPERATORY I ZMIENNE

Do podstawowych operatorów arytmetycznych należą: dodawanie (+), odejmowanie (–), mnożenie (*), dzielenie (/) oraz reszta z dzielenia całkowitego (%).

```
1. #include <iostream>
2. using namespace std;
3.
4. int main() {
5.     cout << 34 / 2 << endl;
6.     cout << 33 / 2 << endl;
7.
8.     return 0;
9. }
```

wynik 17

wynik 16

Podczas dzielenia liczb całkowitych wynik jest liczbą całkowitą.

**Rys. 8.** Podstawowe operacje arytmetyczne w C++ bez zadeklarowanego typu zmiennej

Zmienne przechowują wartości określonego typu, m.in. liczby całkowite, liczby rzeczywiste, wartości logiczne i łańcuchy znaków. Najpierw trzeba zmienną zadeklarować, a potem przypisać jej wartość początkową. Dopiero wtedy można ją modyfikować. Każda zmienna w C++ powinna mieć krótką i znaczącą nazwę (bez polskich liter i spacji), np. **`n`** albo **`liczba`**. Należy przy tym pamiętać, że wielkość liter jest istotna – **`N`** i **`n`** to różne zmienne. Podczas deklaracji należy podać typ danych, jaki będzie przechowywany w zmiennej. Podstawowe typy danych to:

- **`int`** – liczba całkowita,
- **`float`** – liczba rzeczywista,
- **`char`** – pojedynczy znak,
- **`bool`** – wartość logiczna (**`1`** – prawda, **`0`** – fałsz),
- **`void`** – pusty (bez wyniku).

```
1. #include <iostream>
2. using namespace std;
3.
4. int main() {
5.       float a = 33;
6.       float b = 2;
7.       cout << a / b;
8.       return 0;
9. }
```

**wynik** 16.5

**Rys. 9.** Dzielenie w C++ z zadeklarowanymi zmiennymi rzeczywistymi (typu **`float`**)

Aby obliczyć i wyświetlić sumę dwóch liczb, oprócz zmiennych przechowujących wartość składników można zadeklarować również zmienną zawierającą wynik dodawania i wypisać jej wartość za pomocą polecenia **`cout`**.

```
1. #include <iostream>
2. using namespace std;
3.
4. int main() {
5.       int x = 34;
6.       int y = 66;
7.       int wynik = x + y;
8.       cout << wynik;
9.       return 0;
10.}
```

**Rys. 10.** Obliczanie sumy zmiennych całkowitych **`x`** i **`y`**

## Ćwiczenie 2. Podstawowe działania arytmetyczne bez zmiennych

Wyznacz w jednym pliku kolejno: sumę, różnicę, iloczyn, iloraz oraz resztę z dzielenia całkowitego. Do obliczeń wykorzystaj liczby całkowite.

1. Dla każdego przykładu w edytorze kodu źródłowego w funkcji głównej wpisz polecenie **cout** i symbol **<<**.
2. Wpisz odpowiedni wzór oraz **<< endl;**, tak aby wyniki wypisywały się w osobnych wierszach. Zwróć uwagę na kolejność wpisywania liczb.

- Do wyznaczenia sumy wykorzystaj operator +.

```
cout << 34 + 2 << endl;
cout << 2 + 34 << endl;
```

- Do wyznaczenia różnicy wykorzystaj operator –.

```
cout << 17 - 5 << endl;
cout << 5 - 17 << endl;
```

- Do wyznaczenia iloczynu wykorzystaj operator *.

```
cout << 3 * 6 << endl;
cout << 6 * 3 << endl;
```

- Do wyznaczenia ilorazu wykorzystaj operator /.

```
cout << 34 / 2 << endl;
cout << 7 / 3 << endl;
```

- Do wyznaczenia reszty z dzielenia całkowitego wykorzystaj operator %.

```
cout << 19 % 2 << endl;
cout << 14 % 3 << endl;
cout << 15 % 3 << endl;
```

3. Upewnij się, że wzorzec programu zawiera wszystkie potrzebne elementy, w tym polecenie **return 0** i zamykający funkcję nawias klamrowy.
4. Wykonaj obliczenia w pamięci, a następnie skompiluj i uruchom program.

## Ćwiczenie 3. Kupowanie obcej waluty

Oblicz, ile musisz zapłacić w złotych za 126 euro (aktualny kurs wymiany sprawdź na stronie NBP).

1. W edytorze kodu źródłowego w funkcji głównej wpisz polecenie **cout** i symbol **<<**.
2. Wpisz odpowiedni wzór – przemnóż daną kwotę przez aktualny kurs wymiany (pamiętaj, że w C++ separatorem części dziesiętnej i całkowitej jest kropka). Nie zapomnij zakończyć wiersza średnikiem.

- Dla 1 euro = 4,60 zł kod przyjmuje postać:

```
1. #include <iostream>
2. using namespace std;
3.
4. int main() {
5.     cout << 126 * 4.6;
6.     return 0;
7. }
```

❸ Skompiluj i uruchom program.

Operatory porównania wykorzystywane w języku C++ zaprezentowano w poniższej tabeli wraz z przykładami. Warto podkreślić, że wyrażenia logiczne zawsze umieszcza się w nawiasie, a wynikami porównań są wartości logiczne **1** (prawda) lub **0** (fałsz).

| Operatory porównania | | | |
|---|---|---|---|
| == | równe | (3 == 3) → 1 | (4 == 3) → 0 |
| != | różne | (3 != 2) → 1 | (3 != 3) → 0 |
| < | mniejsze | (3 < 4) → 1 | (4 < 3) → 0 |
| <= | mniejsze równe | (4 <= 4) → 1 | (4 <= 3) → 0 |
| > | większe | (4 > 3) → 1 | (3 > 4) → 0 |
| >= | większe równe | (4 >= 4) → 1 | (4 >= 5) → 0 |

## Ćwiczenie 4. Porównanie cen wyrażonych w różnych walutach

Te same buty w Grecji kosztują 49 euro, a w Polsce 200 zł. Sprawdź, czy buty kupione w Grecji są droższe niż buty kupione w Polsce (aktualny kurs wymiany znajdziesz na stronie NBP). Zastosuj operatory arytmetyczne i porównania.

❶ W edytorze kodu źródłowego w funkcji głównej wpisz polecenie **cout** i symbol **<<**.

❷ Wpisz odpowiedni wzór – przemnóż cenę butów w Grecji przez aktualny kurs wymiany (pamiętaj o zastąpieniu przecinka kropką) i porównaj z ceną butów w Polsce. Nie zapomnij zakończyć wiersza średnikiem.

- Dla 1 euro wartego 4,60 zł kod może przyjąć postać:

```
1. #include <iostream>
2. using namespace std;
3.
4. int main() {
5.     cout << (49 * 4.6 >= 200);
6.     return 0;
7. }
```

3. Skompiluj i uruchom program.

- Dla 1 euro wartego 4,60 zł wynikiem jest **1**, czyli prawda, więc buty kupione w Grecji są droższe niż kupione w Polsce.

## Ćwiczenie 5. Średni dystans

W czasie wakacji uczniowie na obozie wędrownym przeszli w ciągu kolejnych trzech dni 12, 14 i 18 km. Zdefiniuj odpowiednie zmienne i oblicz średnią długość pokonywanych tras.

1. W edytorze kodu źródłowego w funkcji głównej zadeklaruj zmienne i przypisz im podane wartości. Ponieważ wynikiem ma być liczba rzeczywista, wskaż typ zmiennej w postaci liczby rzeczywistej. Nie zapomnij zakończyć każdego wiersza średnikiem.

```
5. float a = 12;
6. float b = 14;
7. float c = 18;
```

2. W kolejnym wierszu zadeklaruj zmienną przechowującą średnią podanych wartości i jej typ oraz zapisz wzór, za pomocą którego wyznaczysz tę średnią. Nie zapomnij zakończyć wiersza średnikiem.

```
8. float srednia = (a + b + c) / 3;
```

3. Dodaj polecenie wypisujące wartość zmiennej przechowującej wyznaczony średni dystans.

```
1. #include <iostream>
2. using namespace std;
3.
4. int main() {
5.     float a = 12;
6.     float b = 14;
7.     float c = 18;
8.     float srednia = (a + b + c) / 3;
9.     cout << srednia;
10.    return 0;
11.}
```

4. Sprawdź odpowiedź – skompiluj i uruchom program.

### DEFINIOWANIE FUNKCJI

Funkcje pozwalają podzielić większy problem na mniejsze, względnie niezależne części, a także przyśpieszają i ułatwiają pisanie programu, który staje się dzięki nim bardziej zwięzły i przejrzysty. Definicję umieszcza się powyżej funkcji **`main()`**. Przed nazwą funkcji podaje się typ wyniku funkcji, następnie jej nazwę (bez polskich znaków, spacji) i wstawia nawiasy, w których można umieścić parametr lub parametry. Każdy parametr

poprzedzony jest typem danych. Treść funkcji zawiera się w nawiasach klamrowych. W kolejnych wierszach kodu podaje się instrukcje do wykonania. Polecenie **return** przekazuje szukaną wartość i kończy działanie funkcji.

**ZAPAMIĘTAJ!**

```
<typ wyniku> <nazwa funkcji>(<typ parametru> <parametr>) {
    <instrukcje>
  }
```

```
1. #include <iostream>
2. using namespace std;
3.
4. float pole(float a) {
5.       return a * a;
6. }
7.
8. int main() {
9.      cout << pole(5); // wywołanie funkcji
10.     return 0;
11. }
```

Rys. 11. Przykład obliczania pola kwadratu z wykorzystaniem funkcji

## Ćwiczenie 6. Przeliczanie prędkości

Samolot z Warszawy do Rzymu pokonuje dystans 1320 km. Przeanalizuj dane w tabeli i zdefiniuj funkcję **jak_szybko(float t)**, której parametrem jest czas przelotu samolotu w godzinach, a wynikiem – średnia prędkość tego samolotu.

| Wywołanie funkcji | Wynik |
|---|---|
| **jak_szybko(2.5)** | **528** |
| **jak_szybko(3)** | **440** |

1. Przeanalizuj podane przykłady.
   - Aby wyznaczyć średnią prędkość, należy podzielić drogę przez czas.
2. Nad funkcją **main()** zdefiniuj funkcję **jak_szybko(float t)**, np.

```
4. float jak_szybko(float t) {
5.       return 1320 / t;
6. }
```

```
4. float jak_szybko(float t) {
5.     int s = 1320;
6.     return s / t;
7. }
```

3. Sprawdź działanie programu dla różnych danych.

```
8. cout << jak_szybko(2.5) << endl;
9. cout << jak_szybko(3) << endl;
10. cout << jak_szybko(2.2);
```

4. Skompiluj i uruchom program.

## Ćwiczenie 7. Przeliczanie jednostek

Mila lądowa, jednostka długości stosowana w krajach anglosaskich, jest równa 1,609 km. Przeanalizuj dane w tabeli i zdefiniuj funkcję **`zamiana(float m)`**, której parametrem jest odległość w milach lądowych, a wynikiem – ta sama odległość w kilometrach.

| Wywołanie funkcji | Wynik |
|---|---|
| **`zamiana(10)`** | **`16.09`** |
| **`zamiana(12)`** | **`19.308`** |

1. Przeanalizuj podane przykłady. Aby zamienić mile na kilometry, należy pomnożyć odległość w milach przez 1,609.
2. Zdefiniuj funkcję **`zamiana(float m)`**, a następnie sprawdź jej działanie z różnymi danymi, np.

```
1. #include <iostream>
2. using namespace std;
3.
4. float zamiana(float m) {
5.     return m * 1.609;
6. }
7.
8. int main() {
9.     cout << zamiana(10) << endl;
10.    cout << zamiana(12);
11.    return 0;
12.}
```

3. Skompiluj i uruchom program.

## Ćwiczenie 8. O której godzinie pociąg dotrze do celu

Nocny pociąg z Krakowa do Paryża wyjeżdża ze stacji o godzinie 21.00. Przeanalizuj dane w tabeli i zdefiniuj funkcję **`godzina(int n)`**, której parametrem jest czas przejazdu pociągu w godzinach, a wynikiem – godzina dojazdu do celu.

| Wywołanie funkcji | Wynik |
|---|---|
| **godzina(14)** | **11** |
| **godzina(20)** | **17** |

1. Przeanalizuj podane przykłady i zapisz słownie odpowiednią formułę. 21 + 14 = 35, 21 + 20 = 41. Doba trwa 24 godziny, dlatego aby obliczyć czas przyjazdu pociągu, należy wyznaczyć resztę z dzielenia wyniku dodawania przez 24.
2. W edytorze kodu źródłowego zdefiniuj funkcję **godzina(int n)** – wykorzystaj do obliczeń resztę z dzielenia całkowitego.

```
4. int godzina(int n) {
5.     return (21 + n) % 24;
6. }
```

3. Sprawdź działanie funkcji z różnymi danymi.
4. Skompiluj i uruchom program.

### Ćwiczenie 9. Która godzina będzie w Nowym Jorku?

Sprawdź, jaka strefa czasowa obowiązuje w Nowym Jorku, a następnie przeanalizuj dane w tabeli i zdefiniuj funkcję **godz_nj(int g)**, której parametrem jest pełna godzina w Warszawie, a wynikiem – aktualna pełna godzina w Nowym Jorku.

| Wywołanie funkcji | Wynik |
|---|---|
| **godz_nj(18)** | **12** |
| **godz_nj(2)** | **20** |

## PODSUMOWANIE

- C++ to język kompilowany, za pomocą którego można pisać programy mniej i bardziej skomplikowane. Programy napisane przez człowieka są zamieniane na język maszynowy, dopiero potem mogą być wykonywane przez komputer.
- Główna funkcja w języku C++, w której wywołuje się inne funkcje, nosi nazwę **main()**.
- Definicję funkcji umieszcza się powyżej funkcji **main()**. Przed nazwą funkcji podaje się typ wyniku funkcji, następnie jej nazwę, po czym wstawia się nawiasy, w których można umieścić parametr poprzedzony typem danych. Treść funkcji zawiera się w nawiasach klamrowych. W kolejnych wierszach kodu podaje się instrukcje do wykonania.

- Funkcje mogą przekazywać wynik (polecenie **`return`**) lub powodować określony skutek na ekranie.
- Bloki oznacza się nawiasami klamrowymi, dla większej przejrzystości warto stosować również wcięcia.
- Podstawowe operatory arytmetyczne i porównania są zbliżone do używanych w języku naturalnym, dlatego po napisaniu kilku programów korzystanie z nich staje się intuicyjne.
- Zmienne służą do przechowywania wartości określonego typu, takich jak liczby całkowite i rzeczywiste, wartości logiczne, napisy. Należy im nadawać krótkie i znaczące nazwy.
- Język C++ rozróżnia małe i wielkie litery.
- Rozwiązywany problem warto podzielić na mniejsze problemy, które można niezależnie rozwiązać przez definiowanie funkcji. Gdy problem jest prosty, wystarczy jedna funkcja.

## PYTANIA SPRAWDZAJĄCE

1. Jaka liczba będzie wynikiem operacji **`5 % 2`**?
2. Czym się różni polecenie **`cout << "a";`** od polecenia **`cout << a;`**?
3. Do czego służy polecenie **`return`** w funkcji?

## PYTANIA QUIZOWE

1. Co jest wynikiem działania **`121 % 3`**?
   **a.** 12
   **b.** 1
   **c.** 21
   **d.** 40
2. Co wypisze polecenie **`cout << endl;`**?
   **a.** Spację.
   **b.** Dwie spacje.
   **c.** Cudzysłów.
   **d.** Pusty wiersz.
3. W jaki sposób działa polecenie **`return`**?
   **a.** Wypisuje szukaną wartość.
   **b.** Oblicza wynik funkcji.
   **c.** Przekazuje szukaną wartość – wynik funkcji.
   **d.** Zamienia liczbę na napis.

## ZADANIA DODATKOWE

### Zadanie 1. Która godzina będzie w Warszawie?

Różnica czasu między Los Angeles a Warszawą wynosi dziewięć godzin. Jeżeli w Los Angeles zegar wskazuje godzinę 21.00, to w Warszawie jest godzina 6 rano (następnego dnia). Przeanalizuj dane w tabeli i zdefiniuj funkcję **`godz_waw(int g)`**, której parametrem jest pełna godzina w Los Angeles, a wynikiem – aktualna godzina w Warszawie.



| Wywołanie funkcji | Wynik |
|---|---|
| **`godz_waw(8)`** | **17** |
| **`godz_waw(21)`** | **6** |

### Zadanie 2. Ile to kosztuje?

Zdefiniuj funkcję **`cena(float f)`**, której parametrem jest cena w dolarach funta danego produktu, a wynikiem – cena w złotych kilograma tego produktu. Sprawdź w internecie aktualny kurs dolara oraz przelicznik jednostek masy (funt i kilogram).



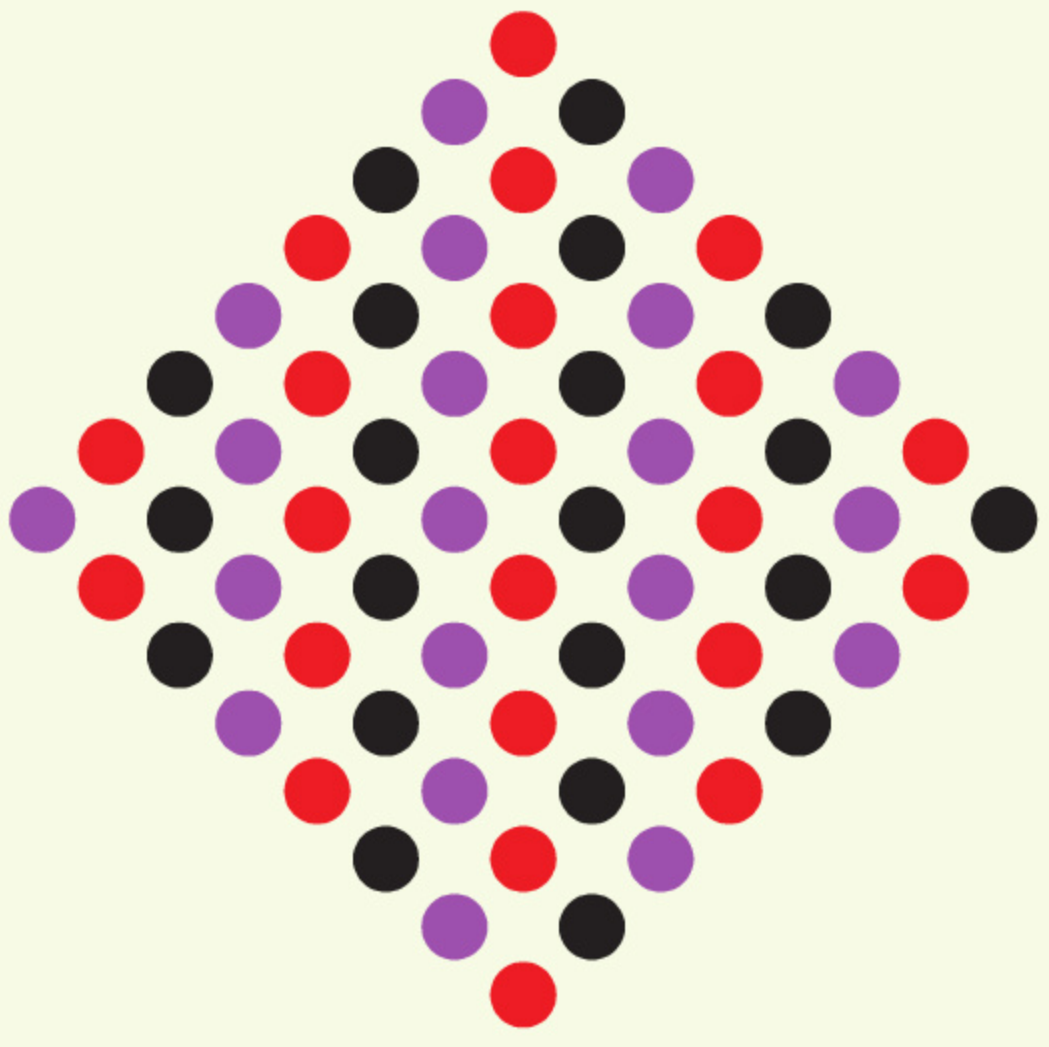
# 15A. Definiowanie funkcji obliczeniowych

- Stosowanie instrukcji warunkowych
- Wykonywanie iteracji za pomocą pętli **for**
- Analizowanie i testowanie rozwiązań

## ZADANIE NA START

### Wzór z korali

Z korali w trzech kolorach można układać ciekawe wzory. Przyjrzyj się poniższemu rysunkowi i określ regułę dotyczącą kolorów korali. Policz, ile jest korali każdego z kolorów, a następnie napisz wzór na ich liczbę w zależności od liczby korali na boku kwadratu.



## ROZWIĄZYWANIE PROBLEMÓW W C++

Podczas rozwiązywania problemów programistycznych należy najpierw postarać się zrozumieć problem, następnie przeanalizować go na kilku przykładach, potem zaproponować sposób wykonania obliczeń i zapisać go zgodnie z regułami obowiązującymi w danym środowisku, wreszcie przetestować.

Zapis formalny zwykle wymaga zdefiniowania funkcji zawierających mniej lub bardziej złożone sekwencje poleceń. W tabeli zamieszczonej na następnej stronie przypomniano konstrukcje stosowane w języku C++, omówione podczas poprzedniej lekcji, oraz zaprezentowano składnię dwóch kolejnych przydatnych konstrukcji – instrukcji warunkowej i pętli **for**.

W tabeli przedstawiono podstawowe konstrukcje stosowane w języku C++.

| Konstrukcja | Składnia | Przykład |
|---|---|---|
| instrukcja **cout** | `cout << argument;` | `cout << "tekst";`<br>`cout << 3;`<br>`cout << x;`<br>`cout << pole_kwadratu(12);` |
| funkcja | `typ_wyniku nazwa(parametry){`<br>`  instrukcje;}` | `int pole_prostokata(int bok1, int bok2){`<br>`  return bok1 * bok2;}` |
| instrukcja warunkowa prosta | `if (warunek)`<br>`  instrukcje;` | `if (x % 3 == 0)`<br>`  cout << "liczba podzielna przez 3";` |
| instrukcja warunkowa złożona | `if (warunek)`<br>`  instrukcje;`<br>`else`<br>`  instrukcje;` | `if (x >= 0)`<br>`  cout << "liczba nieujemna";`<br>`else`<br>`  cout << "liczba ujemna";` |
| pętla **for** | `for (instr_nastart; warunek; instr_kroku)`<br>`  instrukcje;` | `for (i = 0; i < 10; i++)`<br>`  cout << i << endl;` |

## Ćwiczenie 1. Ile korali

Przeanalizuj dane w tabeli i zdefiniuj funkcję **`ile(int k)`**, której parametrem jest liczba korali na boku kwadratu **k**, a wynikiem – łączna liczba korali we wzorze.

| Wywołanie funkcji | Wynik |
|---|---|
| **`ile(5)`** | **25** |
| **`ile(6)`** | **36** |

1. Przeanalizuj treść ćwiczenia i podane przykłady. Możesz wykonać rysunki pomocnicze.



Jeśli bok kwadratu jest zbudowany z 5 korali, to łączna liczba korali tworzących kwadrat wynosi 5 · 5 = 25.



Jeśli bok kwadratu jest zbudowany z 6 korali, to łączna liczba korali tworzących kwadrat wynosi 6 · 6 = 36.

2. Zapisz odpowiednią formułę.

```
k * k
```

3. Zdefiniuj funkcję **ile(int k)**.

```
4. int ile(int k) {
5.     return k * k;
6. }
```

4. Skompiluj i uruchom program. Sprawdź jego działanie z różnymi danymi.

```
1. #include <iostream>
2. using namespace std;
3.
4. int ile(int k) {
5.     return k * k;
6. }
7.
8. int main() {
9.     cout << ile(5) << endl;
10.    cout << ile(6) << endl;
11.    return 0;
12. }
```

## INSTRUKCJA WARUNKOWA

Instrukcja warunkowa pozwala na wykonanie różnych obliczeń w zależności od tego, czy zdefiniowane wyrażenie logiczne jest prawdziwe, czy fałszywe.

- Jeśli liczba jest parzysta, podziel wartość przez 2, w przeciwnym razie odejmij 1.
- Jeśli liczba jest dodatnia, przypisz zmiennej **znak** wartość 1, gdy ujemna, przypisz –1, w przeciwnym razie przypisz 0.

Instrukcja warunkowa może składać się z jednej lub dwóch części.

- Na początku zawsze znajdują się słowo kluczowe **if** oraz warunek lub wyrażenie ujęte w nawiasy okrągłe. Warunek (wyrażenie) może przyjmować wartość **1** lub inną wartość niezerową (prawda) albo **0** (fałsz).
- Potem następuje instrukcja albo ciąg instrukcji do wykonania w przypadku spełnienia warunku. Więcej niż jedną instrukcję należy ująć w nawiasy klamrowe.
- Słowo kluczowe **else** wprowadza instrukcje do wykonania, gdy warunek nie jest spełniony.

**ZAPAMIĘTAJ!**

```
if (<wyrażenie>) {
   <wykonaj instrukcje> }
else {
   <wykonaj instrukcje> }
```

Instrukcja warunkowa na rys. 1 składa się z dwóch części. Można ją odczytać następująco:

- jeśli reszta z dzielenia całkowitego zmiennej **x** przez 2 jest równa 0, wyświetl połowę tej liczby;
- w przeciwnym razie wyświetl liczbę o 1 mniejszą.

```
1. if (x % 2 == 0)
2.     x = x / 2;
3. else
4.     x = x - 1;
```

**Rys. 1.** Przykładowa instrukcja warunkowa złożona z dwóch elementów

Instrukcja warunkowa na rys. 2 składa się z trzech części. Można ją odczytać następująco:

- jeśli zmienna **x** jest większa od 0, przypisz zmiennej **zn** wartość 1;
- jeśli poprzedni warunek jest fałszywy i zmienna **x** jest mniejsza od 0, przypisz zmiennej **zn** wartość –1;
- w przeciwnym razie przypisz zmiennej **zn** wartość 0.

```
1. if (x > 0)
2.     zn = 1;
3. else if (x < 0)
4.     zn = -1;
5. else
6.     zn = 0;
```

**Rys. 2.** Przykładowa instrukcja warunkowa złożona z trzech elementów

Jeśli w ramach instrukcji warunkowej jest wykonywana tylko jedna instrukcja, nie trzeba używać nawiasów klamrowych. W przypadku większej liczby instrukcji nawiasy są konieczne.

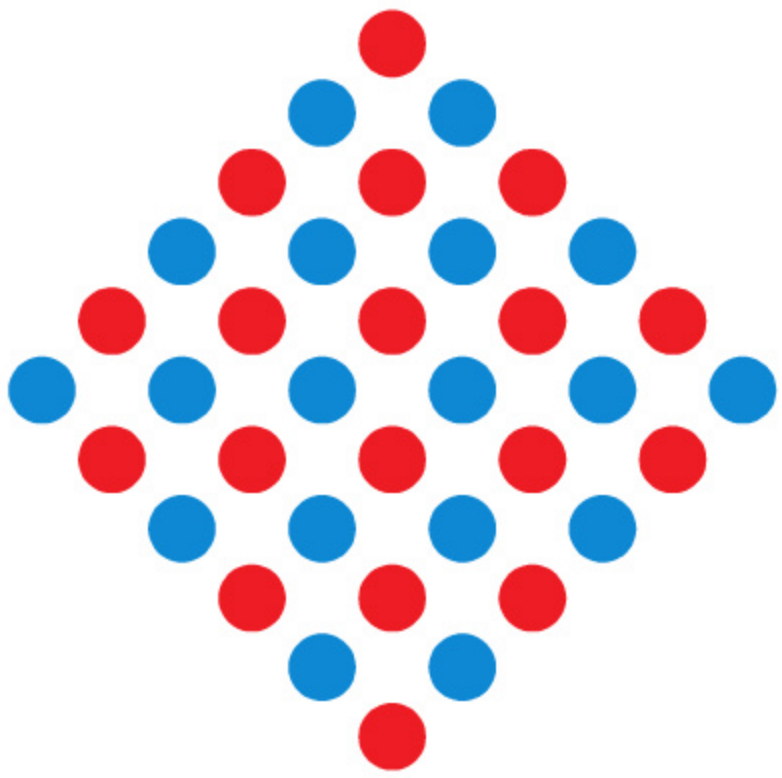
## Ćwiczenie 2. Dwukolorowy romb

Przeanalizuj dane w tabeli i zdefiniuj funkcję **`ile_kolor(int k, char kolor)`**, której parametrami są liczba korali na boku kwadratu **`k`** i kolor korali – **`c`** (czerwony) lub **`n`** (niebieski), a wynikiem jest liczba korali podanego koloru we wzorze.

| Wywołanie funkcji | Wynik |
|---|---|
| `ile_kolor(6, 'c')` | **18** |
| `ile_kolor(7, 'c')` | **24** |
| `ile_kolor(6, 'n')` | **18** |
| `ile_kolor(7, 'n')` | **25** |

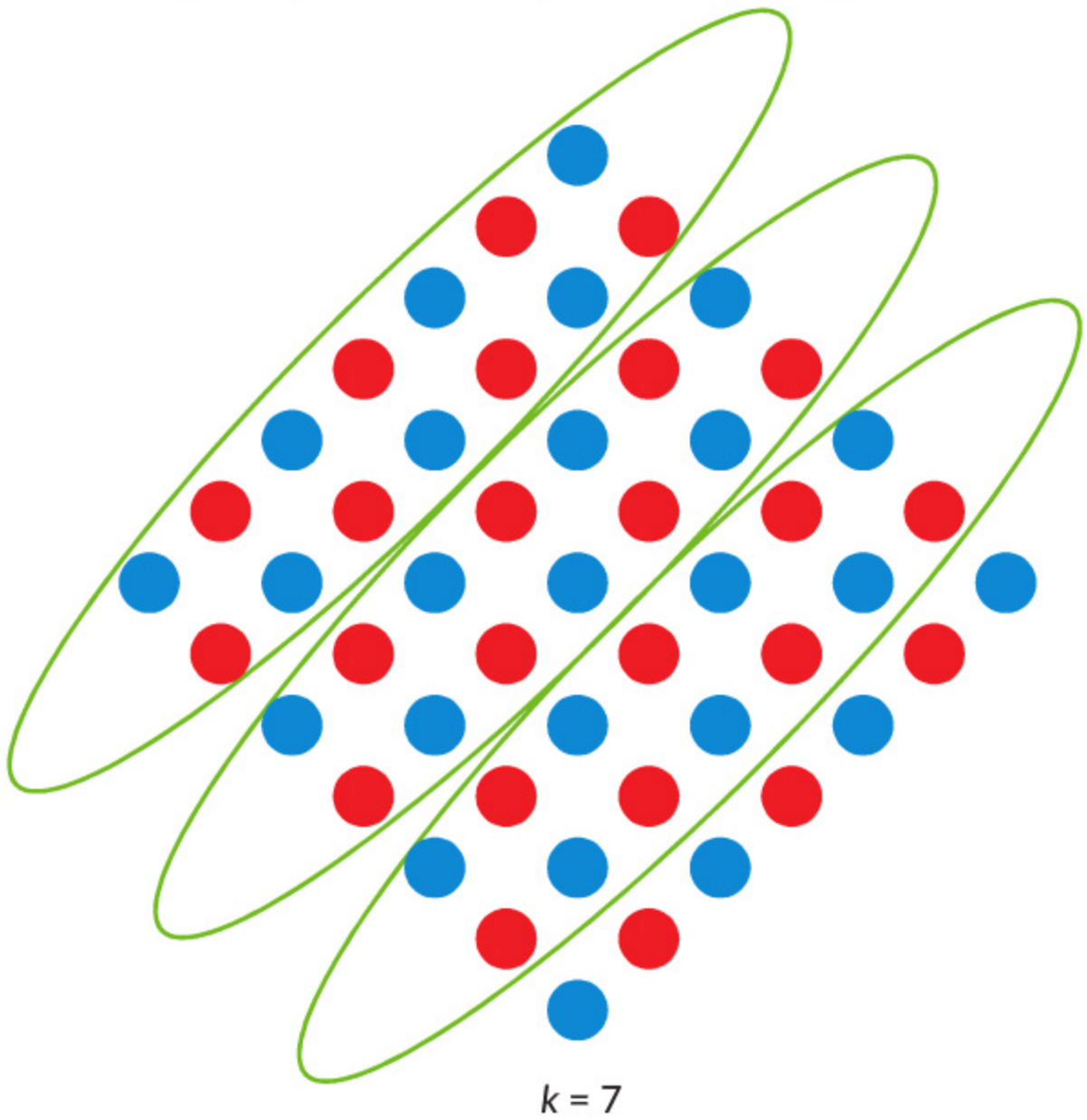
1. Przeanalizuj treść ćwiczenia i przykłady. Możesz wykonać rysunki pomocnicze.

- Dla wartości parzystych parametru **k** liczba korali dla obu kolorów jest jednakowa. Aby wyznaczyć liczbę korali w danym kolorze, trzeba pomnożyć przez siebie liczbę korali tworzących bok i podzielić przez 2: **6 * 6 / 2 = 18**.



*k* = 6

- Dla wartości nieparzystych parametru **k** liczba korali czerwonych jest o 1 mniejsza od liczby korali niebieskich. Aby wyznaczyć liczbę korali w danym kolorze, trzeba wprowadzić dwa warunki:
  - jeśli liczone są korale czerwone, wynikiem będzie liczba korali tworzących bok kwadratu, pomnożona przez siebie i podzielona przez 2: **7 * 7 / 2**;
  - w przeciwnym razie (tj. dla korali niebieskich) wynikiem będzie liczba korali tworzących bok kwadratu, pomnożona przez siebie i pomniejszona o 1, a następnie podzielona przez 2 i powiększona o 1.



*k* = 7

Połącz w pary dwa sąsiednie wiersze z wyjątkiem ostatniego, a następnie wyznacz osobno liczbę korali czerwonych i niebieskich w każdej parze wierszy i w wierszu ostatnim.

Korale czerwone:
(3 + 4) + (3 + 4) + (3 + 4) + 3 = 24.
Zapis, którego podstawę stanowi liczba 7:
24 = 7 * 7 / 2.
Zapis uogólniony: k * k / 2.

Korale niebieskie:
(4 + 3) + (4 + 3) + (4 + 3) + 4 = 25.
Zapis, którego podstawę stanowi liczba 7:
25 = (7 * 7 - 1) / 2 + 1.
Zapis uogólniony: (k * k – 1) / 2 + 1.

2. Zapisz odpowiednie formuły.

- Liczba korali czerwonych:

```
k * k / 2
```

- Liczba korali niebieskich:

```
(k * k - 1) / 2 + 1
```

3. Zdefiniuj funkcję **`ile_kolor(int k, char kolor)`**. Jeśli funkcja ma więcej niż jeden parametr, oddzielamy je przecinkiem.

```
8.  int ile_kolor(int k, char kolor) {
9.      if (kolor == 'c')
10.         return k * k / 2;
11.     else
12.         return (k * k - 1) / 2 + 1;
13. }
```

4. Skompiluj i uruchom program. Sprawdź jego działanie z różnymi danymi, w tym z parzystym i nieparzystym rozmiarem wzoru.

## PĘTLA FOR

Instrukcja iteracji służy do powtarzania jakiejś instrukcji lub bloku instrukcji w programie, np. dodawania wielu liczb do siebie. Dla każdego elementu sekwencji jest wykonywany podany ciąg instrukcji.

**ZAPAMIĘTAJ!**

```
for (stan_początkowy; warunek; instrukcja przy każdym obrocie pętli) {
   <instrukcje>
 }
```

Lista instrukcji wykonuje się tak długo, jak warunek jest prawdziwy. Przy każdym wykonaniu pętli zmienna sterująca pętlą (zwyczajowo stosuje się nazwę **`i`**) zostaje zmieniona zgodnie z instrukcją – w przypadku pętli na rys. 2 zapis **`i++`** oznacza zwiększenie wartości zmiennej o **`1`** (zapis **`i++`** jest równoznaczny z zapisem **`i+=1`**, czyli **`i = i + 1`**).

```
1. for (int i = 0; i < 5; i++)
2.     cout << i << endl;
```

Ponieważ w tej pętli jest wykonywana tylko jedna instrukcja, nie trzeba używać nawiasów klamrowych. W przypadku większej liczby instrukcji nawiasy są konieczne.

**wynik**

```
0
1
2
3
4
```

**Rys. 2.** Pętla wypisująca liczby od 0 do 4

## Ćwiczenie 3. Piramida

Przeanalizuj dane w tabeli i zdefiniuj funkcję **`piramida(int k)`**, której parametrem jest liczba korali w podstawie **`k`**, a wynikiem – łączna liczba korali.

| Wywołanie funkcji | Wynik |
|---|---|
| **`piramida(5)`** | **15** |
| **`piramida(6)`** | **21** |

Zauważ, że ćwiczenie można wykonać na dwa sposoby.

### Sposób 1

1. Przeanalizuj treść ćwiczenia i podane przykłady.

- Rozpatrz liczbę korali i zapamiętaj ją na zmiennej o nazwie **`licznik`**. Najpierw **`licznik`** jest równy 0. Potem zwiększa się o 1, czyli o liczbę korali w najwyższym rzędzie. W dalszej kolejności zwiększa się o 2, tj. o liczbę korali w rzędzie drugim od góry. Potem o 3...



1 + 2 + 3 + 4 + 5 = 15

1 + 2 + 3 + 4 + 5 + 6 = 21

2. Zapisz odpowiednie formuły.

```
licznik = 0
licznik = licznik + 1
licznik = licznik + 2
…
licznik = licznik + k
```

3. Przeanalizuj powyższy zapis i go uogólnij – zastosuj pętlę.

```
3. for (int i = 1; i < k + 1; i++)
4.     licznik = licznik + i;
```

4. Zdefiniuj funkcję **`piramida(k)`**.

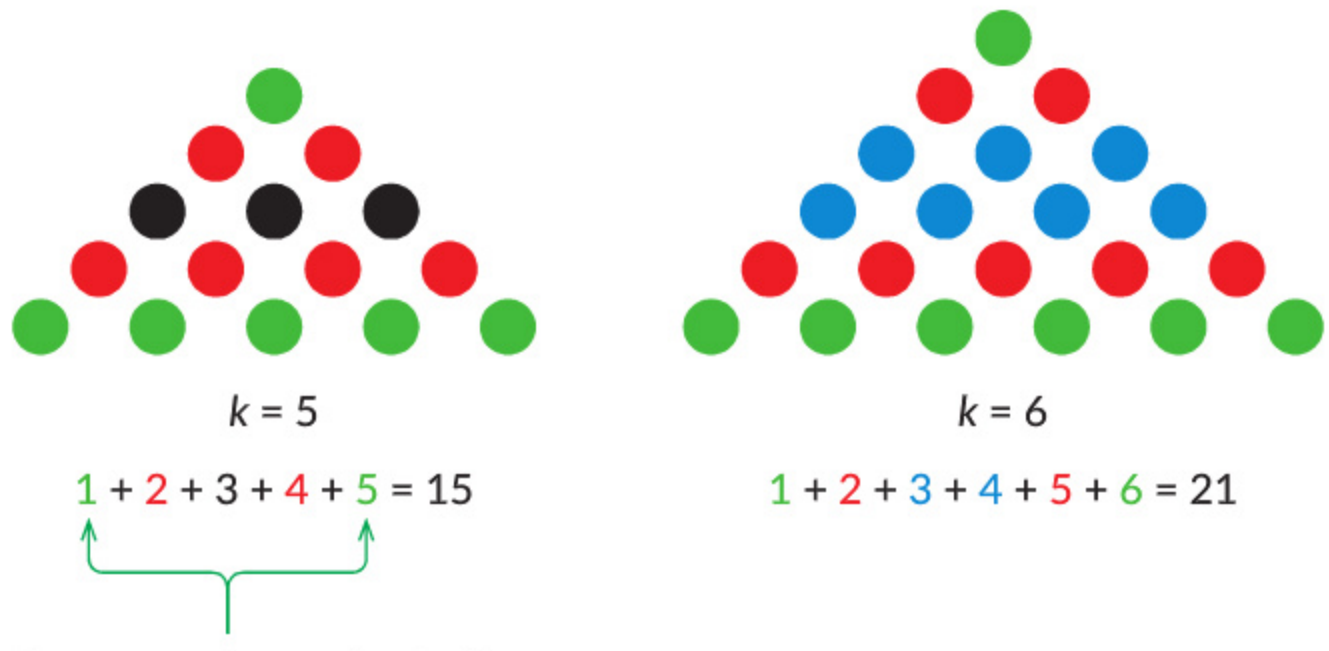
```
1. int piramida(int k) {
2.     int licznik = 0;
3.     for (int i = 1; i < k + 1; i++)
4.         licznik = licznik + i;
5.     return licznik;
6. }
```

5. Skompiluj i uruchom program. Sprawdź jego działanie z różnymi danymi.

### Sposób 2

1. Przeanalizuj treść ćwiczenia i podane przykłady.

- Zauważ, że rzędy można pogrupować parami – pierwszy z ostatnim, drugi z przedostatnim itd.




2. Zapisz ogólną zależność wzorem. Ponieważ *k* albo *k* + 1 jest zawsze liczbą parzystą, zastosuj dzielenie całkowite.

```
k * (k + 1) / 2
```

3. Zdefiniuj funkcję **`piramida(int k)`**.

```
1. int piramida(int k) {
2.     return k * (k + 1) / 2;
3. }
```

4. Skompiluj i uruchom program. Sprawdź jego działanie z różnymi danymi.

## Ćwiczenie 4. Piramida – porównanie funkcji

Porównaj działanie obu funkcji z ćwiczenia 3 dla różnych parametrów.

1. Zapisz funkcje **`piramida1(int k)`** oraz **`piramida2(int k)`** w jednym pliku, a następnie skonstruuj polecenie, które wypisze na ekranie wyniki dla obu funkcji.

- Kod wypisujący wyniki dla parametrów w zakresie od 1 do 19 może przyjąć postać:

```
1. for (int i = 1; i < 20; i++)
2.     cout << piramida1(i) << " " << piramida2(i) << endl;
```

2. Podsumuj swoje spostrzeżenia. Zastanów się, który wariant analizy problemu był dla ciebie bardziej czytelny. Czy przejście od rozpoznania prawidłowości do uogólnienia nie sprawiło ci trudności? Jakie dostrzegasz zalety i wady każdej z przyjętych strategii?

### Ćwiczenie 5. Trzy kolory

Zaprojektuj własny wzór złożony z koralików w co najmniej trzech kolorach. Napisz funkcję, której wynikiem będzie liczba korali każdego koloru.

## PODSUMOWANIE

- Instrukcja warunkowa pozwala na wykonanie różnych obliczeń w zależności od tego, czy zdefiniowane wyrażenie logiczne jest prawdziwe (ma wartość **`1`** lub inną wartość niezerową), czy fałszywe (ma wartość **`0`**).
- Instrukcja warunkowa może składać się z jednej lub dwóch części. Na początku znajdują się słowo kluczowe **`if`** oraz warunek. Potem następuje instrukcja lub ciąg instrukcji do wykonania w razie spełnienia warunku. Słowo kluczowe **`else`** wprowadza instrukcję do wykonania, gdy warunek nie jest prawdziwy.
- Instrukcja iteracji służy do powtarzania jakiejś instrukcji lub bloku instrukcji w programie, np. dodawania wielu liczb do siebie. Dla każdego elementu sekwencji wykonuje się podany ciąg instrukcji.
- W pętli **`for`** zwykle używa się zmiennej sterującej pętlą o nazwie **`i`**.

## PYTANIA SPRAWDZAJĄCE

1. Czy w instrukcji warunkowej zawsze występuje słowo kluczowe **`else`**? Odpowiedź uzasadnij.
2. Jak w języku C++ zapisać instrukcję iteracji, aby zmienna sterująca pętlą zmieniała wartości od 10 do 0 włącznie?
3. Z jakimi wartościami parametrów warto testować każde zadanie?

## PYTANIA QUIZOWE

1. Jaka ostatnia wartość zostanie wypisana po wykonaniu poniższej pętli?

```
1. for (int i = 3; i < 11; i = i + 1)
2.     cout << i << endl;
```

**a.** 8
**b.** 9
**c.** 10
**d.** 11

2. Z jakim argumentem należy wywołać funkcję, aby uzyskać wynik 4?

```
int test(int x) {
    if (x % 2 == 0)
        return  x / 2;
    else
        return x - 1;
}
```

**a.** 5 lub 6.
**b.** 6 lub 8.
**c.** 5 lub 8.
**d.** 5, 6 lub 8.

3. Jaką wartość będzie miała zmienna **licznik** po wykonaniu poniższych poleceń?

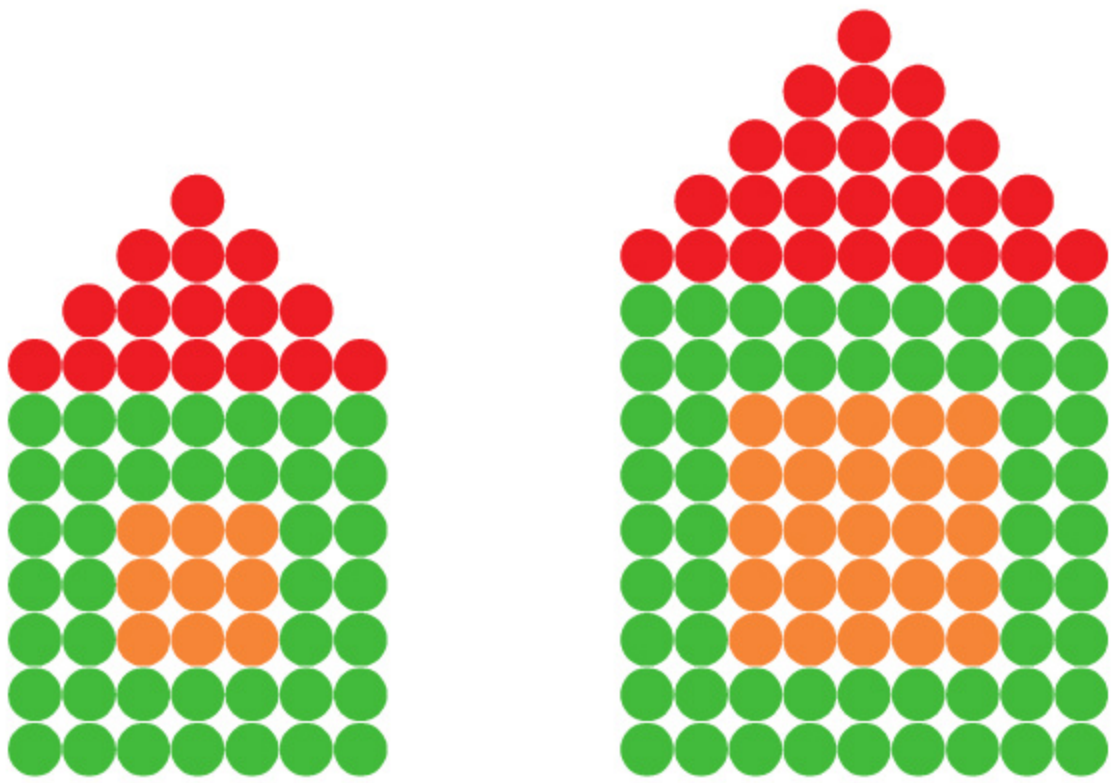
```
1. licznik = 3;
2. licznik = licznik + 3;
```

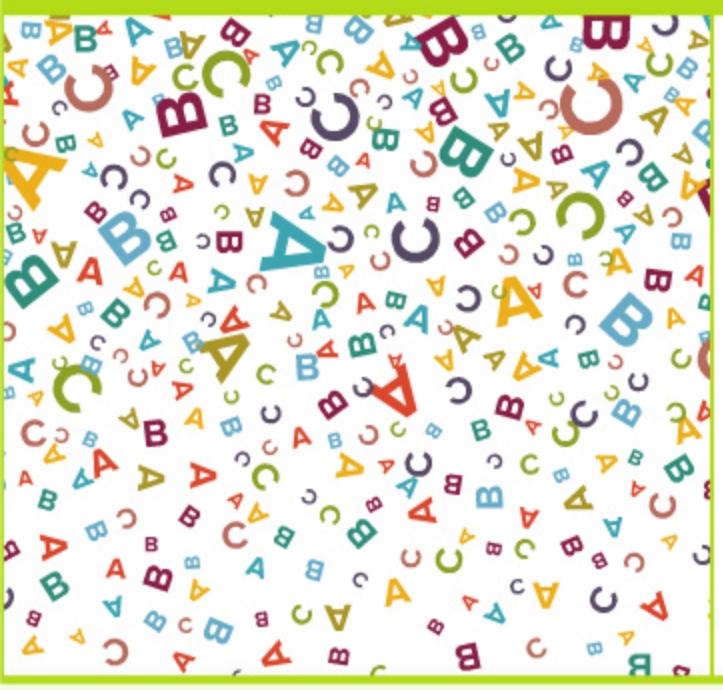
**a.** 3
**b.** 6
**c.** 9
**d.** 12

## ZADANIE DODATKOWE

### Zadanie 1. Trzykolorowe domki

Napisz wzór i zdefiniuj funkcję, która będzie wyliczać liczbę korali danego koloru.





# 16A. Wyszukiwanie wzorca w tekście

- Podstawowe operacje na napisach
- Algorytm naiwny wyszukiwania wzorca w tekście
- Porównywanie i przeszukiwanie napisów

## ZADANIE NA START

**Jakie symbole można znaleźć w DNA?**

DNA to kwas deoksyrybonukleinowy. Tworzą go nukleotydy zbudowane z deoksyrybozy, jednej grupy fosforanowej oraz jednej zasady azotowej: adeniny (A), guaniny (G), cytozyny (C) lub tyminy (T). Przykładowy łańcuch DNA, który zawiera symbole A, C, G i T, to TCTAAAGATATCGGG. Długość tego łańcucha wynosi 15. Policz, ile razy symbole A, C, G i T występują w łańcuchu TCTAACAGCCCCATATCGGG.



## NAPISY

Napis to ciąg znaków. Mogą się na niego składać litery, cyfry oraz inne znaki, np. spacja. W języku C++ napisy oznacza się podwójnym cudzysłowem (**" "**), a deklaruje się je jako typ **`string`**. W odróżnieniu od napisów pojedyncze znaki (cyfry, małe i wielkie litery, znaki interpunkcyjne itp.), które deklaruje się jako zmienną typu **`char`**, zapisuje się w pojedynczych cudzysłowach (apostrofach).

Aby wypisać napis, należy zastosować polecenie **`cout`**.

**ZAPAMIĘTAJ!**

```
string <zmienna> = "<ciąg znaków>";
cout << <zmienna>;
```

Na rys. 1 pokazano przykład deklaracji i wypisywania zmiennych typu **`string`**, których wartością są napisy.

```
1. string s1 = "biologia";
2. string s2 = "molekularna";
3. cout << s1 << endl;
4. cout << s2 << endl;
```

**Rys. 1.** Deklarowanie i wypisywanie zmiennych

## DODAWANIE NAPISÓW

Napisy można łączyć ze sobą – jest to tzw. konkatenacja. W C++ ciągi znaków dodaje się za pomocą znaku +.

**ZAPAMIĘTAJ!**

```
string <zmienna1> = "<ciąg znaków>";
string <zmienna2> = "<ciąg znaków>";
cout << <zmienna1> + <zmienna2>;
```

W wierszu czwartym na rys. 2 dodawane są trzy napisy, środkowy napis jest spacją.

```
1. string s1 = "biologia";
2. string s2 = "molekularna";
3. cout << s1 + s2 << endl;
4. cout << s1 + " " +  s2 << endl;
```

**Rys. 2.** Konkatenacja napisów

## WYZNACZANIE DŁUGOŚCI NAPISU

Aby wyznaczyć liczbę znaków zadeklarowanej zmiennej, należy wykorzystać metodę **size()**.

**ZAPAMIĘTAJ!**

```
<zmienna>.size()
```

Na rys. 3 przedstawiono przykład zmiennej **napis**, której przypisano wartość będącą sumą trzech napisów (**s1**, spacja i **s2**), oraz funkcji obliczającej długość tej zmiennej (liczbę znaków).

```
1. string s1 = "biologia";
2. string s2 = "molekularna";
3. string napis = s1 + " " + s2;
4. cout << napis.size() << endl;
```

**Rys. 3.** Wypisywanie liczby znaków zmiennej **napis**

## ANALIZOWANIE NAPISU

Do pojedynczego znaku napisu należy odwoływać się przez indeks. Jego wartość, umieszczona po nazwie zmiennej w nawiasie kwadratowym i liczona od zera, wskazuje odpowiedni znak. Jeśli zmiennej **napis** przypisze się wartość **"genetyka"**, to **napis[0]** będzie oznaczać **'g'**, **napis[4]** będzie oznaczać **'t'**, a **napis[7]** – **'a'**.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| g | e | n | e | t | y | k | a |

Analizę napisu litera po literze można wykonać również za pomocą pętli **for**. Na początku zmiennej sterującej **i** przypisywana jest wartość 0, w każdym obrocie pętli jej wartość zwiększana jest o 1 (**i++**), a pętla jest wykonywana tak długo, jak długo warunek **i < <napis>.size()** jest spełniony.

**ZAPAMIĘTAJ!**

```
for (i = 0; i < <napis>.size(); i++)
    cout << <napis>[i] << endl;
```

Oto przykład wyświetlania napisu **"genetyka"** litera po literze.

```
1. string napis = "genetyka";
2. for (int i = 0; i < napis.size(); i++)
3.     cout << napis[i] << endl;
```

**Rys. 4.** Wypisywanie kolejnych liter zmiennej **napis**

## Ćwiczenie 1. Zliczanie pojedynczych znaków w tekście

Przeanalizuj dane w tabeli i zdefiniuj funkcję **ile(string s, char z)**, której parametrami są ciąg znaków oraz znak A, C, G lub T, a wynikiem jest liczba wystąpień danego znaku w ciągu znaków.

| Wywołanie funkcji | Wynik |
|---|---|
| **ile("CAATAAAAA", 'T')** | **1** |
| **ile("TCTAAAGATATCGGG", 'A')** | **5** |

1. Przeanalizuj treść ćwiczenia i podane przykłady.
   - Zlicz wystąpienia kolejnych liter w ciągu znaków **"CAATAAAAA"**: litera A występuje 7 razy, litera C – raz, litera G nie występuje, litera T pojawia się raz.
   - Zlicz wystąpienia kolejnych liter w ciągu znaków **"TCTAAAGATATCGGG"**: litera A występuje 5 razy, litera C – 2 razy, litera G – 4 razy, litera T – 4 razy.
2. Przemyśl możliwe rozwiązania problemu, zapisz formuły, a następnie zdefiniuj funkcję.

- Aby zliczyć wystąpienia danego znaku w napisie, należy zadeklarować zmienną pomocniczą **ile** (wartość początkowa 0), która będzie przechowywała liczbę wystąpień szukanego symbolu w danym napisie. Algorytm realizujący to zadanie polega na analizowaniu napisu znak po znaku. W przypadku napotkania znaku **z** wartość zmiennej **ile** jest zwiększana o 1.

```
1.  for (int i = 0; i < s.size(); i++)
2.      if (s[i] == z)
3.          ile++;
```

3. Zdefiniuj funkcję **ile(string s, char z)**.

```
1. int ile(string s, char z) {
2.     int ile = 0;
3.     for (int i = 0; i < s.size(); i++)
4.         if (s[i] == z)
5.             ile++;
6.     return ile;
7. }
```

4. Skompiluj i uruchom program. Sprawdź działanie funkcji z różnymi danymi, w tym z napisami złożonymi z jednakowych liter.

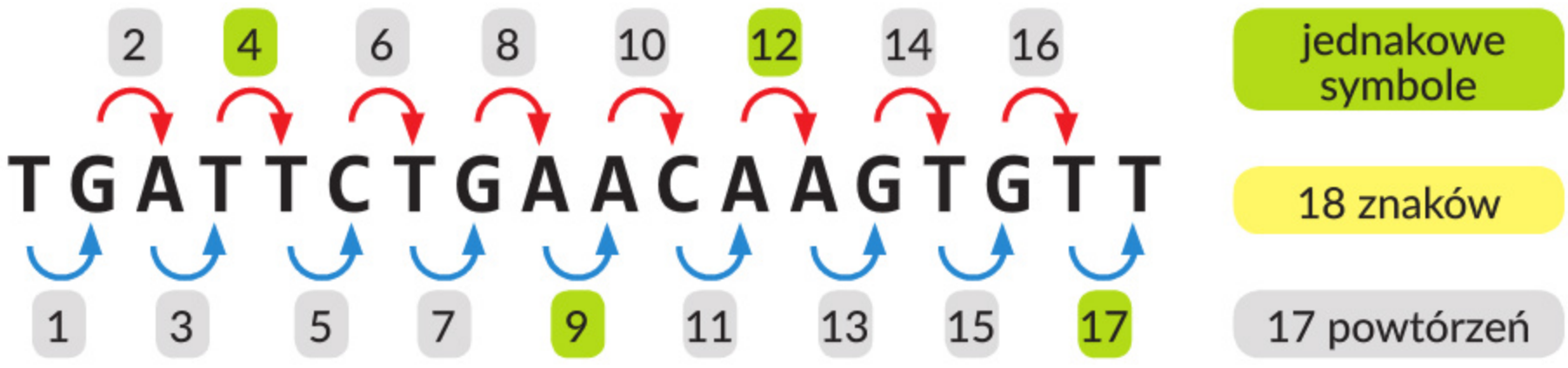
## Ćwiczenie 2. Zliczanie ciągów znaków w tekście

Przeanalizuj dane w tabeli i zdefiniuj funkcję **pary(string s)**, której parametrem jest łańcuch DNA, a wynikiem – liczba wystąpień par jednakowych symboli (np. w ciągu znaków TTAT jest jedna taka para, w TTTA – 2, w TATA – 0).

| Wywołanie funkcji | Wynik |
|---|---|
| **pary("TGATTCTGAACAAGTGTT")** | **4** |
| **pary("TGATATTCGATGTGAAAAAGTCATACTGTT")** | **6** |

1. Przeanalizuj treść ćwiczenia i podane przykłady.

- Wykonaj rysunek pomocniczy.




Aby wyszukać jednakowe symbole, trzeba przeanalizować cały napis znak po znaku. Porównuje się pary, czyli bieżący znak z następnym, np. TG, GA, AT, TT itd. Pętla będzie wykonywana o 1 raz mniej, niż wynosi długość napisu.

❷ Zapisz kolejne formuły – przy konstruowaniu warunku zastosuj indeksowanie. Sprawdź, czy znak o indeksie **i** jest równy znakowi o indeksie **i + 1**.

```
s[i] == s[i + 1]
```

❸ Zdefiniuj funkcję **pary(string s)**.

```
1. int pary(string s) {
2.     int ile = 0;
3.     for (int i = 0; i < s.size() - 1; i++)
4.         if (s[i] == s[i + 1])
5.             ile++;
6.     return ile;
7. }
```

❹ Skompiluj i uruchom program. Sprawdź działanie funkcji z różnymi danymi.

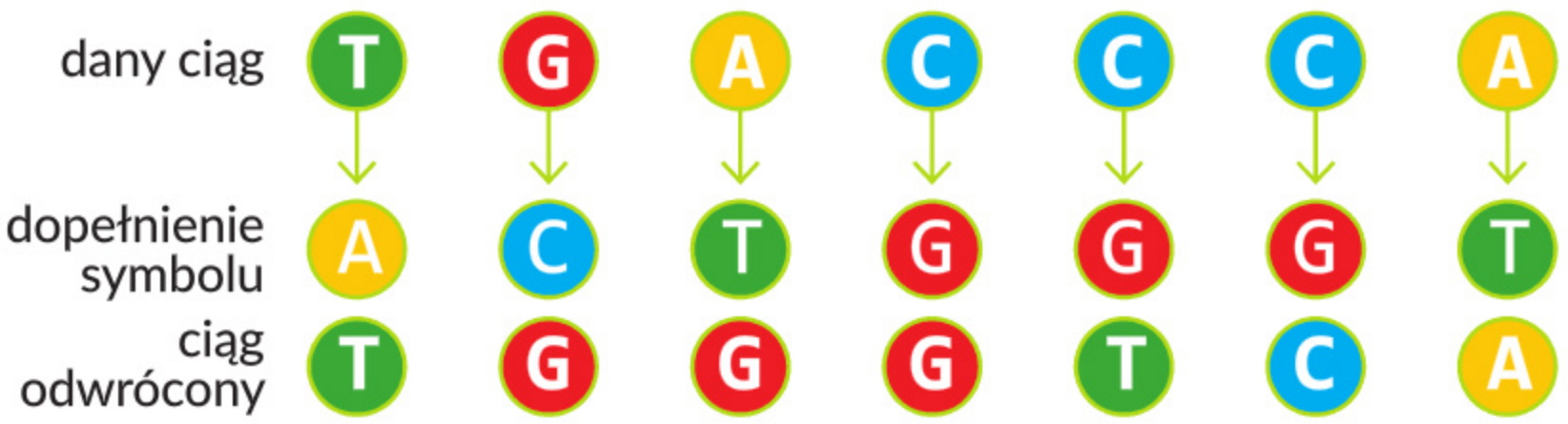
## Ćwiczenie 3. Zmiana struktury ciągu znaków

Odwrotne uzupełnienie łańcucha DNA to ciąg utworzony przez odwrócenie danego ciągu i dopełnienie każdego symbolu, tj. zamiany A na T i odwrotnie oraz C na G i odwrotnie (np. CTGA → AGTC → TCAG). Przeanalizuj dane w tabeli i zdefiniuj funkcję **oduz(string s)**, której parametrem jest łańcuch DNA, a wynikiem – jego odwrotne uzupełnienie.

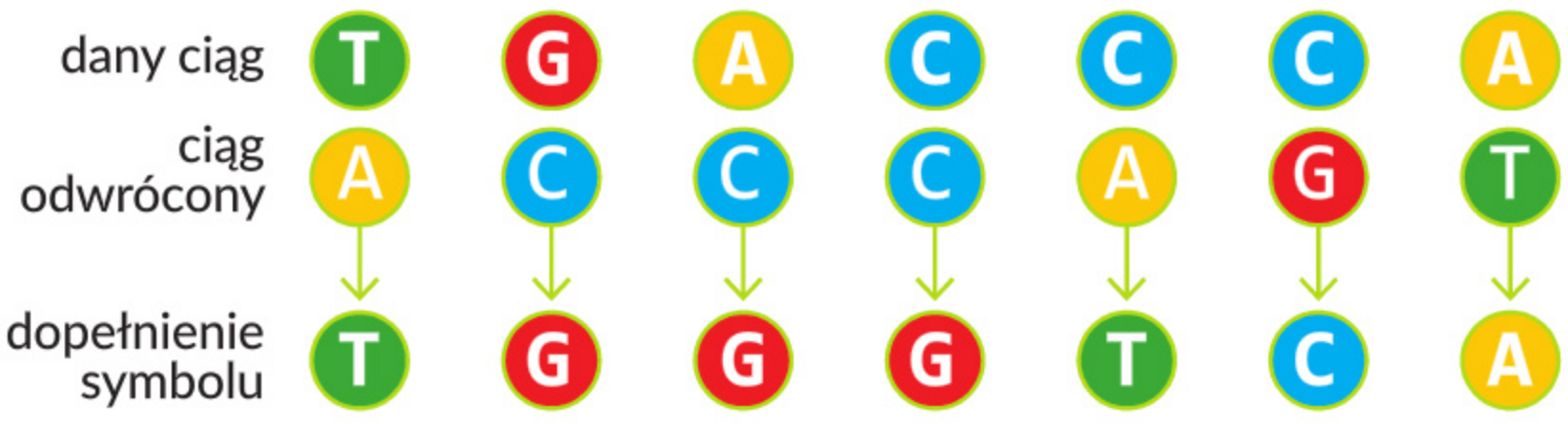
| Wywołanie funkcji | Wynik |
|---|---|
| **oduz("TGACCCA")** | **"TGGGTCA"** |
| **oduz("TGATATTCGATGTGAAAAA")** | **"TTTTTCACATCGAATATCA"** |

❶ Przeanalizuj treść ćwiczenia i podane przykłady.

- Dla danego ciągu TGACCCA musisz utworzyć jego dopełnienie (zamiana litery A na T, T na A, C na G i G na C) – wynikiem będzie ACTGGGT. Następnie trzeba go odwrócić, czyli zapisać od końca – wynikiem będzie TGGGTCA. Wykonaj rysunek pomocniczy.




- Kolejność operacji (zamiana i odwracanie) nie ma znaczenia.



2. Zapisz kolejne formuły.

- Najpierw nadaj wartość początkową zmiennej **t**, która będzie wynikiem. Wynik ma być napisem, więc wartość początkowa będzie napisem pustym ("").

```
t = ""
```

- Ponieważ w C++ nie da się zmodyfikować napisu – zmienić lub zastąpić w nim poszczególnych znaków – utwórz nową zmienną do przechowywania przekształconego łańcucha DNA, odwróć ciąg i zamień parami symbole, z uwzględnieniem kolejności napisów w dodawaniu.

```
if (s[i] == 'A')
    zn = 'T'
else if (s[i] == 'T')
    zn = 'A'
else if (s[i] == 'C')
    zn = 'G'
else
    zn = 'C'
```

- Algorytm polega na przeglądaniu danego napisu. W trakcie przeglądania, gdy kolejnym symbolem będzie A, to zmiennej pomocniczej **zn** algorytm nada wartość T – i odwrotnie; gdy symbolem będzie C, to zmiennej pomocniczej **zn** nada wartość G – i odwrotnie. Do wyniku algorytm doda wartość zmiennej pomocniczej **zn** – w ten sposób odwróci się przekształcany napis.

```
t = zn + t
```

3. Zdefiniuj funkcję **oduz(string s)**.

```
1.  string oduz(string s) {
2.      string t = "";
3.      char zn;
4.      for (int i = 0; i < s.size(); i++) {
5.          if (s[i] == 'A')
6.              zn = 'T';
7.          else if (s[i] == 'T')
8.              zn = 'A';
9.          else if (s[i] == 'C')
10.             zn = 'G';
11.         else
12.             zn = 'C';
13.         t = zn + t;
14.     }
15.     return t;
16. }
```

4. Skompiluj i uruchom program. Sprawdź działanie funkcji z różnymi danymi.

# ALGORYTM

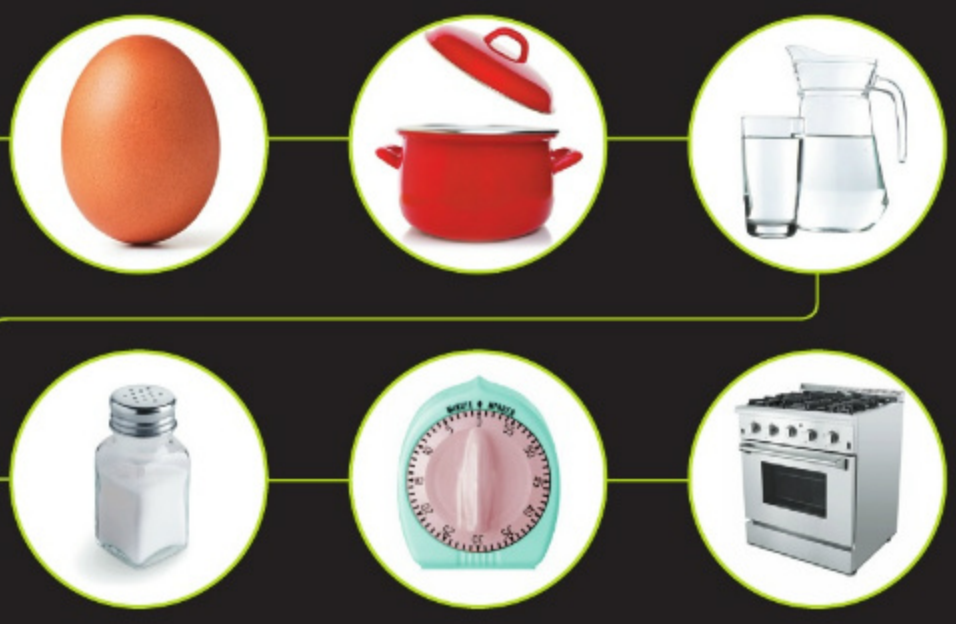
**Algorytm to przepis postępowania prowadzący do rozwiązania problemu. Określa dane wejściowe oraz ciąg czynności, które należy w tym celu wykonać. Wykonawcami algorytmu mogą być i komputer, i człowiek.**

**1.** **Określenie problemu:**
zwięzłe, precyzyjne

Chcę zjeść jajko na miękko na śniadanie.

**2.** **Zdefiniowanie modeli i pojęć:**
opisanie danych wejściowych i zależności wyniku od danych

Co jest potrzebne?



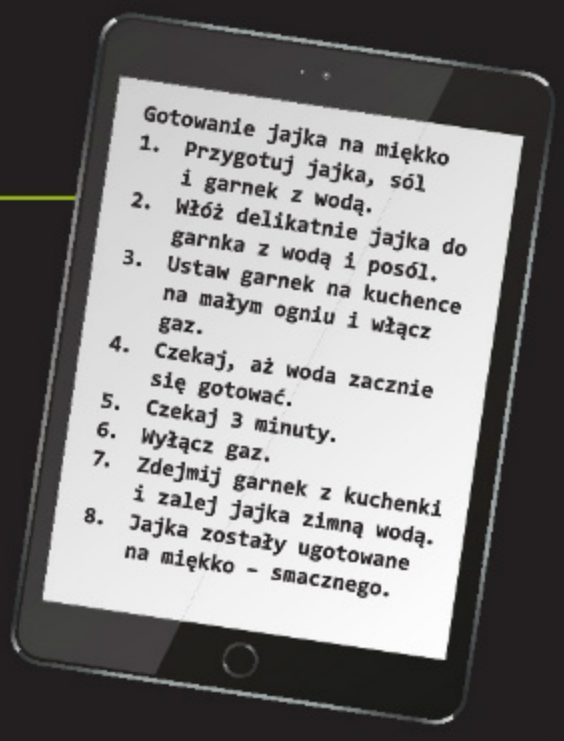
**3.** **Znalezienie rozwiązania:**
opisanie możliwych rozwiązań i wybór najlepszego

Metody gotowania



**4.** **Zaprogramowanie rozwiązania:**
napisanie kodu programu

Przepis




**5.** **Testowanie rozwiązania:**
dla różnych danych spełniających przyjęte założenia

Śniadanie



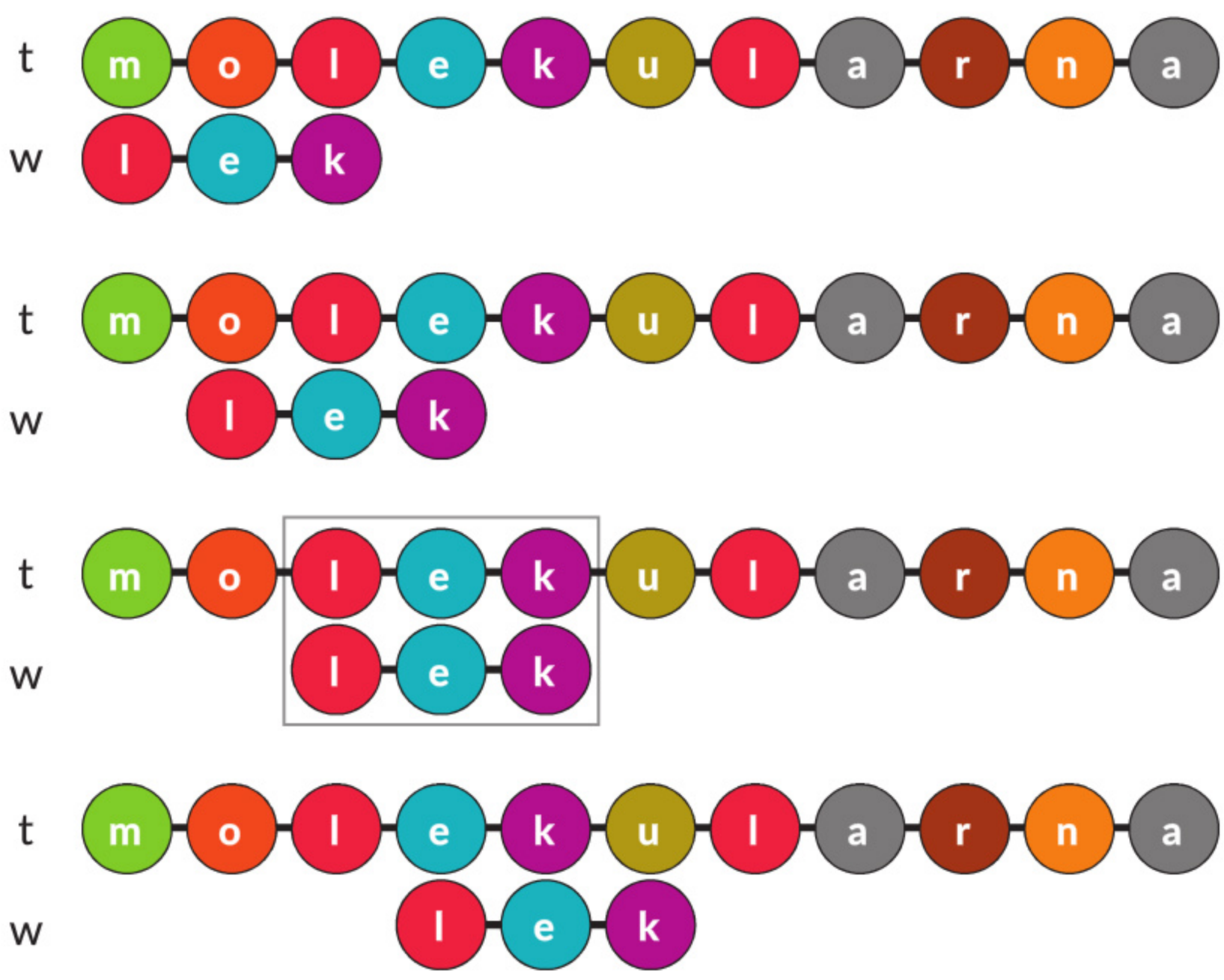
### Ćwiczenie 4. Zmiana struktury ciągu znaków

Przeanalizuj dane w tabeli i zdefiniuj funkcję **`ta(string s)`**, której parametrem jest łańcuch DNA, a wynikiem – łańcuch DNA z gwiazdką (*) w miejscu symboli C, G i T.

| Wywołanie funkcji | Wynik |
|---|---|
| `ta("TGACCCA")` | `"**A***A"` |
| `ta("TGATATTCGATGTGAAAAA")` | `"**A*A****A****AAAAA"` |

## ALGORYTM NAIWNY WYSZUKIWANIA WZORCA W TEKŚCIE

Wzorzec to spójny ciąg znaków, który występuje w przeszukiwanym tekście – jest nim np. fraza **`lek`** występująca w słowie **`molekularna`**. Algorytmy umożliwiające wyszukiwanie wzorca w danym skończonym łańcuchu znaków są wykorzystywane m.in. w edytorach tekstu, przeglądarkach internetowych oraz programach pocztowych. Za pomocą wzorców można też określić, z czego ma się składać adres e-mail. Sekwencja znaków zadana do odszukania może się powtarzać albo w ogóle nie występować w analizowanym tekście.




**Rys. 5.** Cztery z dziewięciu etapów procesu wyszukiwania wzorca w słowie **`molekularna`**

Na powyższym rysunku wzorzec został odnaleziony na trzecim etapie procesu – wszystkie trzy litery są zgodne. Jeżeli chcesz znaleźć wszystkie wystąpienia wzorca w tekście, to algorytm kontynuuje przeszukiwanie tekstu od następnego znaku aż do momentu, gdy liczba nieprzebadanych znaków stanie się mniejsza od długości wzorca.

**ZAPAMIĘTAJ!**

**Algorytm naiwny wyszukiwania wzorca** (inaczej – algorytm N) to najprostszy algorytm, który polega na porównywaniu kolejnych liter tekstu (`t`) i wzorca (`w`), począwszy od pierwszej litery każdego z nich. W razie niezgodności wzorzec przesuwa się o jedną pozycję w prawo i algorytm bada drugi znak tekstu i pierwszy znak wzorca. Jeśli znaki są zgodne, porównywane są następne znaki.

### Ćwiczenie 5. Przeszukiwanie DNA

Dany jest łańcuch DNA o sekwencji TCTAACAGCCCCATATCGGG. Na którym etapie algorytmu N zostanie odnaleziony wzorzec CAT? Ile wzorców zostanie znalezionych?

## PODSUMOWANIE

- Algorytmika i programowanie mają zastosowanie w różnych dziedzinach nauki, m.in. w biologii molekularnej, a także w codziennym funkcjonowaniu w społeczeństwie cyfrowym.
- Napis, czyli ciąg znaków zdefiniowany jako zmienna i ujęty w cudzysłowy, wypisuje się za pomocą polecenia **cout**.
- Konkatenację (łączenie) napisów umożliwia znak dodawania.
- Długość ciągu znaków wyznacza się za pomocą metody `size()`.
- Aby przeanalizować napis znak po znaku, należy zastosować pętlę **for** z indeksowaniem.
- Napisy, czyli ciągi znaków, mają wiele wbudowanych funkcji, które można wykorzystać do rozwiązywania problemów.
- Chociaż w rozbudowanych systemach stosuje się bardziej zaawansowane algorytmy wyszukiwania, które działają szybciej, do prostych zastosowań wystarczy algorytm naiwny.
- Algorytm naiwny wyszukiwania wzorca w tekście polega na porównywaniu kolejnych liter tekstu i wzorca, począwszy od pierwszej litery każdego z nich. W razie niezgodności wzorzec przesuwa się o jedną pozycję w prawo i algorytm bada drugi znak tekstu oraz pierwszy znak wzorca. Jeśli znaki są zgodne, porównywane są następne znaki.

## PYTANIA SPRAWDZAJĄCE

1. Jakie podstawowe operacje można wykonać na napisach?
2. W jaki sposób można w C++ wypisać napis znak po znaku?
3. Ile razy trzeba porównywać dwa znaki, aby znaleźć w tekście o długości `n` wzorzec o długości `w`?

## PYTANIA QUIZOWE

1. Jaka wartość zostanie wypisana dla napisu `w = "wiaderko"` w wyniku polecenia `cout << w.size();`?
   **a.** 5
   **b.** 10
   **c.** 12
   **d.** 6
2. Jaka litera zostanie wypisana dla napisu `w = "kaktus"` w wyniku działania polecenia `cout << w[5] << endl;`?
   **a.** `k`
   **b.** `s`
   **c.** `u`
   **d.** `a`
3. Ile razy wykona się poniższa pętla dla napisu `w = "laseczek"`?

```
for (int i = 0; i < w.size(); i++)
    cout << w[i] << endl;
```

   **a.** 8
   **b.** 10
   **c.** 2
   **d.** 7

## ZADANIE DODATKOWE

### Zadanie 1. Sekwencja błędów

Odległość Hamminga między dwoma łańcuchami DNA oznacza liczbę symboli, którymi różnią się oba łańcuchy. Odległość Hamminga między łańcuchami **`s`** i **`t`** o tej samej długości wynosi 8. Niepasujące symbole są zaznaczone kolorem:

- łańcuch **`s`** – CAGGCTACTACCGGTAT,
- łańcuch **`t`** – CATCGTAATGCAGGGCT.

Przeanalizuj dane w tabeli i zdefiniuj funkcję **`hamming(string s, string t)`**, gdzie **`s`** i **`t`** to łańcuchy o tej samej długości. Wynikiem funkcji jest liczba będąca odległością Hamminga między danymi łańcuchami.

| Wywołanie funkcji | Wynik |
|---|---|
| `hamming("CAGGTT", "CAGGTT")` | `0` |
| `hamming("CAGGCTACTACCGGTAT", "CATCGTAATGCAGGGCT")` | `8` |



# 17A. Przetwarzanie napisów

- Wyodrębnianie fragmentu napisu
- Szyfry przestawieniowe
- Sprawdzanie, czy napis jest palindromem

## ZADANIE NA START

**Czy umiesz odczytać poniższe sentencje**

Odczytaj poniższe zdania, a następnie opisz metodę szyfrowania, jaką zastosowano w każdym przypadku.

patetyr śoin e wimra ęejzdneai
tytepa einśor w ęraim ainezdej
aqpdebtayity raoośynoiuel wy meixabrlęt jvewdfzgeinpiaaq
apeytt roinśe w mraię jndeieza

dgiz erdawr bąąit maw óiyrl ceą
eizdg awrd ąibąr mat yróiw ącel
godczaiseg dwrownad rląwbqiyąx tsaqmj wmicópriyh ltejcnąj
gidze dwra rbiąą tam wóriy lecą

eb zrpca yin eamk łocayz
zeb ycarp ein am yzcałok
bveuzm pqriapcoyl nbipek mhau kcoałlahctzmyz
bez parcy nie ma kłczoay




## WYODRĘBNIANIE FRAGMENTU NAPISU

Podczas przetwarzania napisów można odwoływać się do poszczególnych znaków za pomocą indeksów. Umożliwia to wyodrębnianie fragmentów napisów. Indeksy to liczby z zakresu od 0 do liczby o 1 mniejszej od długości napisu. Za pomocą nawiasów kwadratowych uzyskuje się dostęp do pojedynczego znaku – wystarczy podać jego indeks.

```
1. string s = "szyfrowanie";
2. cout << s[0] << endl;      // s
3. cout << s[9] << endl;      // i
```

**Rys. 1.** Wypisanie znaków o indeksie 0 i 9

## SZYFRY PRZESTAWIENIOWE

Kodowanie wiadomości może się odbywać przez przestawianie znaków. W tak zaszyfrowanym tekście występują wszystkie znaki z tekstu jawnego, ale w innej kolejności. Kolejność liter zmieniana jest według określonego schematu, np. pisanie wspak.

**ZAPAMIĘTAJ!**

**Szyfrogram** to tekst, który został zaszyfrowany.

**Tekst jawny** to tekst, który nie został zaszyfrowany lub został odszyfrowany.

Nauka, która zajmuje się szyfrowaniem i deszyfrowaniem wiadomości, nosi nazwę **kryptografii**.

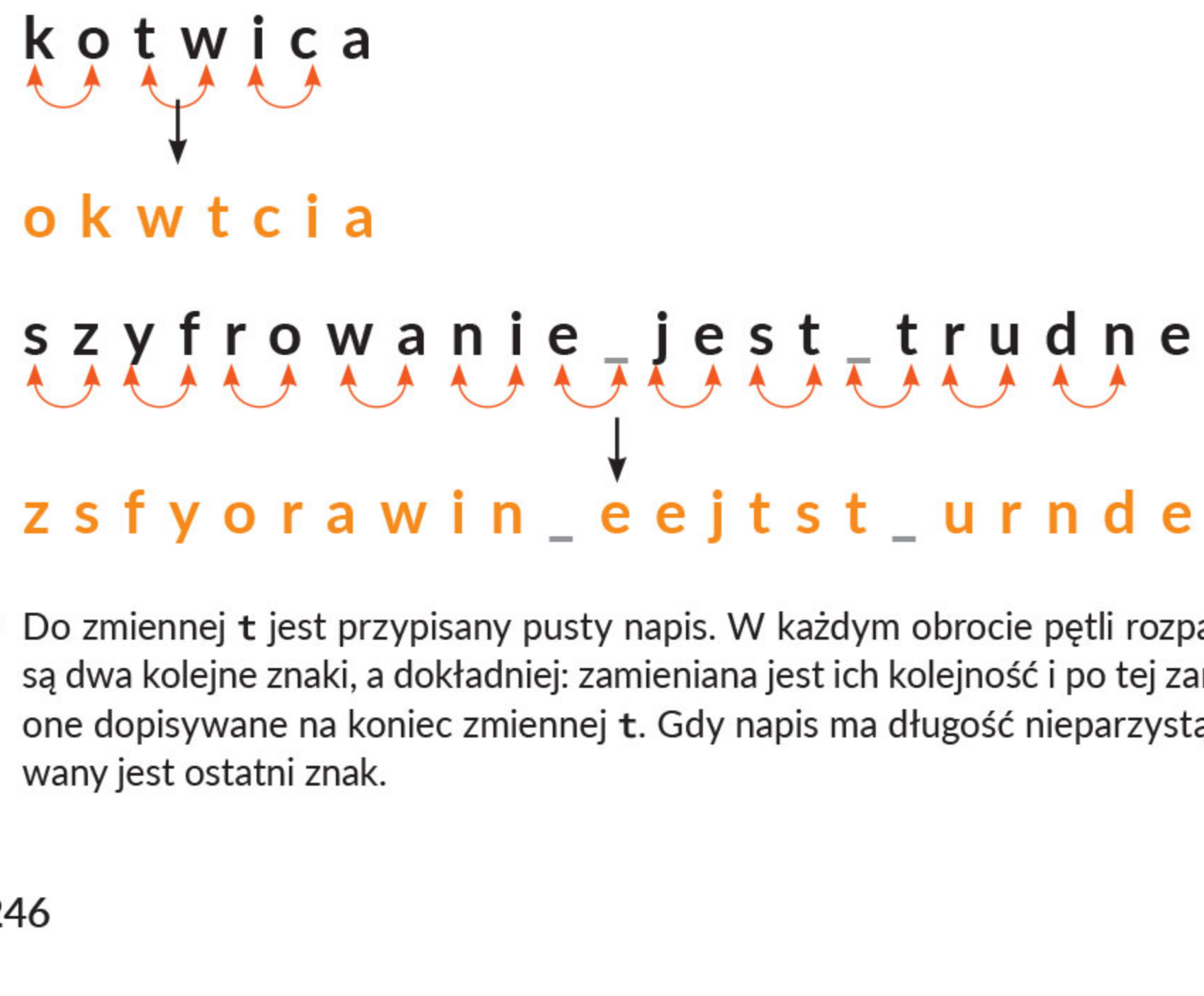
### Ćwiczenie 1. Przestawianie parami

Przeanalizuj dane w tabeli i zdefiniuj funkcję **`szyfr(string s)`**, której parametrem jest tekst jawny składający się z małych liter alfabetu łacińskiego i pojedynczych spacji, a wynikiem – szyfrogram utworzony przez przestawienie kolejnych par znaków, przy czym spacja jest traktowana jak każda inna litera i również podlega zamianie.

| Wywołanie funkcji | Wynik |
|---|---|
| `szyfr("kotwica")` | `"okwtcia"` |
| `szyfr("szyfrowanie jest trudne")` | `"zsfyorawin eejtst urnde"` |

1. Przeanalizuj treść ćwiczenia i podane przykłady.

- Rozrysuj algorytm krok po kroku.




- Do zmiennej **`t`** jest przypisany pusty napis. W każdym obrocie pętli rozpatrywane są dwa kolejne znaki, a dokładniej: zamieniana jest ich kolejność i po tej zamianie są one dopisywane na koniec zmiennej **`t`**. Gdy napis ma długość nieparzystą, dopisywany jest ostatni znak.

2. Zapisz kolejne formuły.

- Zastosuj pętlę **for**, aby przeglądać dany tekst co dwa znaki i dopisywać do zmiennej pary znaków zamienione kolejnością w stosunku do danych.

```
for (int i = 0; i < s.size() - 1; i += 2)
     t = t + s[i + 1] + s[i]
```

- Dopisanie ostatniego znaku to instrukcja wykonywana po zakończeniu pętli.

```
if (s.size() % 2 != 0)
    t = t + s[s.size() - 1]
```

3. Zdefiniuj funkcję **szyfr(string s)**.

```
1. string szyfr(string s) {
2.     string t = "";
3.     for (int i = 0; i < s.size() - 1; i += 2)
4.         t = t + s[i + 1] + s[i];
5.     if (s.size() % 2 != 0)
6.         t = t + s[s.size() - 1];
7.     return t;
8. }
```

3. Sprawdź działanie funkcji z różnymi danymi, w tym z napisami o parzystej i nieparzystej długości.

- Dlaczego ważne jest, żeby testować funkcję z napisami zarówno parzystej, jak i nieparzystej długości?

### PARKAN

Choć *Encyklopedia Britannica* opisuje parkan jako szyfr popularny w szkołach, to osoba niemająca z nim wcześniej do czynienia musi się sporo natrudzić, zanim odczyta ukrytą wiadomość. Prosta wersja szyfru polega na zapisaniu liter tekstu jawnego w dwóch wierszach. Na początek usuwa się wszystkie spacje – to częsta praktyka w kryptografii, ponieważ podział na wyrazy sprawia, że złamanie szyfru staje się łatwiejsze. Następnie należy w pierwszym wierszu zapisać litery na pozycjach nieparzystych, w drugim – na pozycjach parzystych. Aby utworzyć szyfrogram, trzeba odczytać tekst wierszami. Oczywiście, przy szyfrowaniu parkanem nie trzeba ograniczać się do dwóch rzędów.

| Tekst jawny | Szyfrowanie | Szyfrogram |
|---|---|---|
| albumy | a b m<br>l u y | abmluy |
| domek | d m k<br>o e | dmkoe |

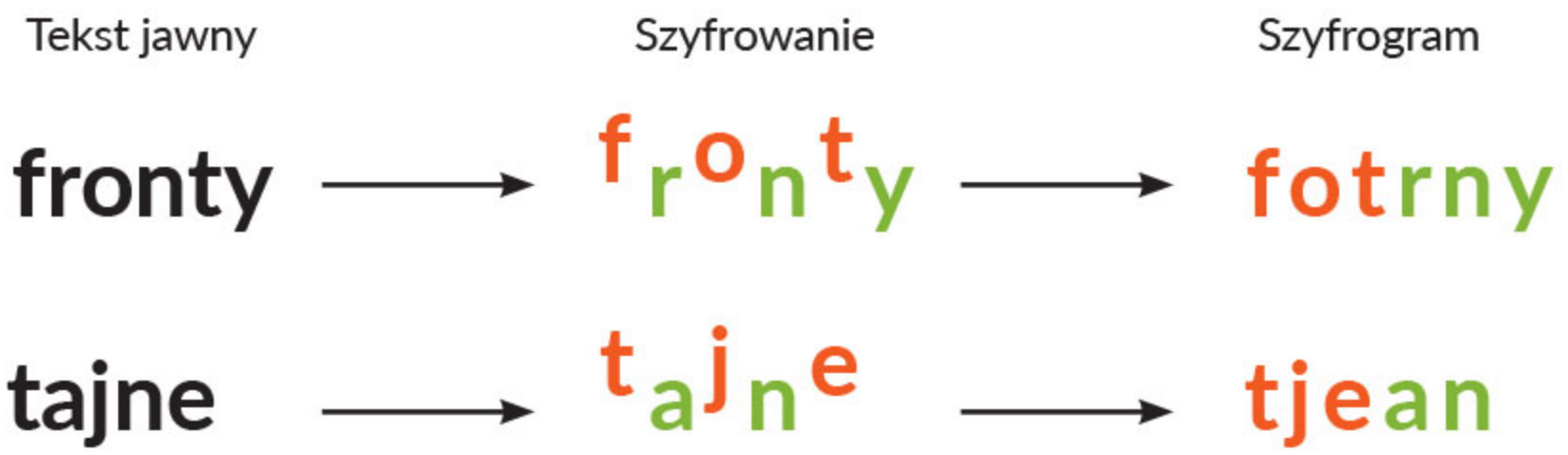
**Rys. 2.** Szyfrowanie parkanem

## Ćwiczenie 2. Parkan

Przeanalizuj dane w tabeli i zdefiniuj funkcję **parkan(string s)**, której parametrem jest tekst jawny złożony z małych liter alfabetu łacińskiego, a wynikiem – szyfrogram utworzony za pomocą szyfru parkan.

| Wywołanie funkcji | Wynik |
|---|---|
| **parkan("fronty")** | **"fotrny"** |
| **parkan("tajne")** | **"tjean"** |

1. Przeanalizuj treść ćwiczenia i podane przykłady.
   - Wykonaj rysunki pomocnicze.




2. Opracuj algorytm – do pustego słowa najpierw trzeba dopisać znaki zaznaczone kolorem pomarańczowym, a potem zielonym.
3. Zdefiniuj funkcję **parkan(string s)**.

```
1. string parkan(string s) {
2.     string w = "";
3.     for (int i = 0; i < s.size(); i += 2)
4.         w = w + s[i];
5.     for (int i = 1; i < s.size(); i += 2)
6.         w = w + s[i];
7.     return w;
8. }
```

4. Sprawdź działanie funkcji z różnymi danymi.

### PALINDROM

Słowa, zdania i liczby identyczne niezależnie od tego, czy są czytane od lewej do prawej, czy od prawej do lewej, np. kajak, „A to kanapa pana kota", 18081, XIX, to palindromy (z gr. biec z powrotem). Wyrażenia tego typu tworzono już w starożytności. Nie wiadomo, czy stanowiło to tylko zabawę, czy też element obrządków magicznych albo rodzaj kodu. Klasyczna metoda sprawdzania, czy dane wyrażenie jest palindromem, polega na porównywaniu pierwszego znaku z ostatnim, drugiego z przedostatnim itd. Wobec tego dla wyrażenia złożonego z siedmiu znaków wykonuje się trzy porównania (tak jak dla wyrażenia złożonego z sześciu znaków), gdyż środkowego znaku nie ma potrzeby porównywać.

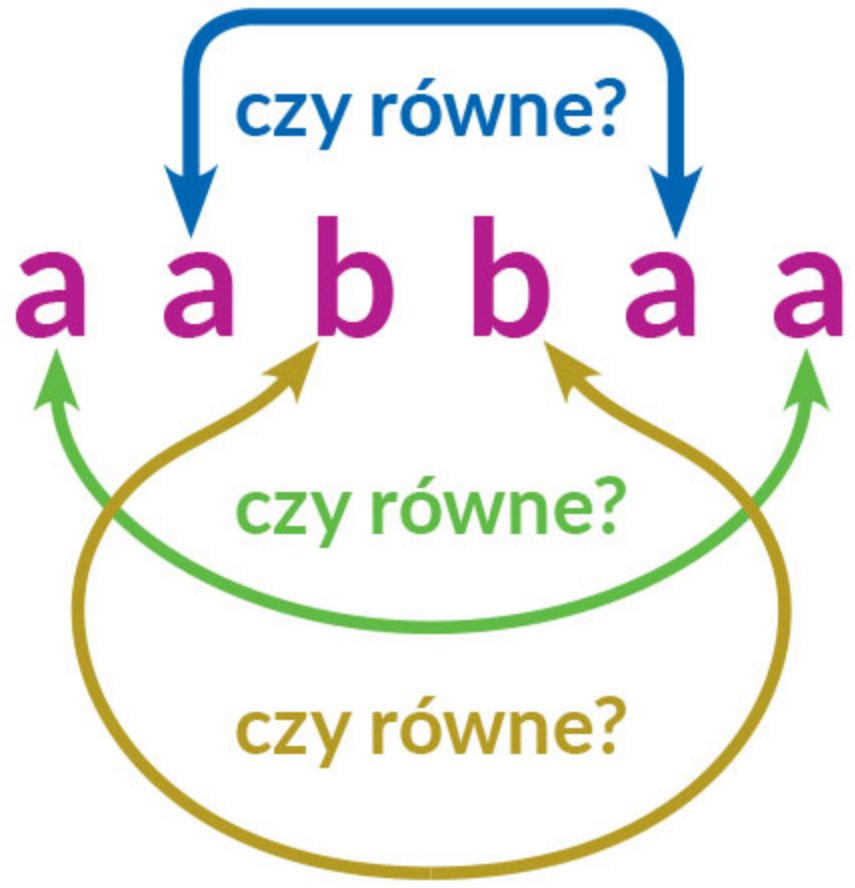
## Ćwiczenie 3. Czy to jest palindrom

Przeanalizuj dane w tabeli i zdefiniuj funkcję logiczną **`czy_palindrom(string s)`**, której parametrem jest napis zawierający tylko małe litery alfabetu łacińskiego, a wynikiem – wartość logiczna **`1`** (prawda), gdy napis jest palindromem, lub **`0`** (fałsz), gdy nim nie jest.

| Wywołanie funkcji | Wynik |
|---|---|
| `czy_palindrom("aabbaa")` | `1` |
| `czy_palindrom("aaxyaa")` | `0` |

1. Przeanalizuj treść ćwiczenia i podane przykłady.

- Wykonaj rysunek pomocniczy.




2. Opracuj algorytm, który polega na sprawdzeniu znaków parami. Najpierw bada, czy pierwsza litera jest identyczna z ostatnią, a jeśli tak, to kontynuuje sprawdzanie kolejnych dwóch liter (drugiej i przedostatniej itd.). Jeśli litery nie są identyczne, kończy działanie funkcji i otrzymuje w wyniku **`0`** (fałsz).

3. Zdefiniuj funkcję **`czy_palindrom(string s)`**. Wykorzystaj typ logiczny **`bool`**, który przyjmuje wartość **`1`** (prawda) lub **`0`** (fałsz).

```
1. bool czy_palindrom(string s) {
2.     int dl = s.size();
3.     for (int i = 0; i < dl / 2; i++)
4.         if (s[i] != s[dl - 1 - i])
5.             return 0;
6.     return 1;
7. }
```

4. Sprawdź działanie funkcji z różnymi danymi, zwłaszcza z napisami, które są palindromami.

## Ćwiczenie 4. Czy zdanie jest palindromem

Przeanalizuj dane w tabeli i zdefiniuj funkcję logiczną **`czy_palindrom2(string s)`**, której parametrem jest napis zawierający małe litery i spacje, a wynikiem – wartość logiczna **`1`** (prawda), gdy napis jest palindromem, lub **`0`** (fałsz), gdy nim nie jest.

| Wywołanie funkcji | Wynik |
|---|---|
| `czy_palindrom2("a to kanapa pana kota")` | `1` |
| `czy_palindrom2("a to kanapka pana kota")` | `0` |

1. Przeanalizuj treść ćwiczenia i podane przykłady.

- Zauważ, że ćwiczenie można rozwiązać w dwóch krokach – najpierw pomiń spacje, potem zastosuj funkcję z ćwiczenia 3.

2. Zdefiniuj funkcję **`pomin(string s)`**, która usuwa spacje z tekstu.

```
1. string pomin(string s) {
2.     string pom = "";
3.     for (int i = 0; i < s.size(); i++)
4.         if (s[i] != ' ')
5.             pom = pom + s[i];
6.     return pom;
7. }
```

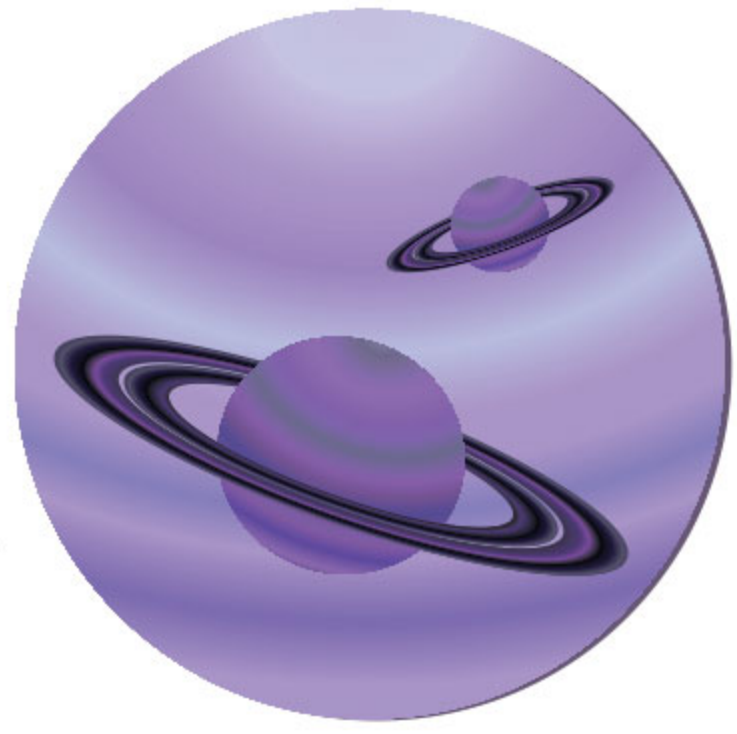
3. Zdefiniuj funkcję **`czy_palindrom2(string s)`** z wykorzystaniem funkcji z ćwiczenia 3 **`czy_palindrom(string s)`**.

```
1. bool czy_palindrom2(string s) {
2.     return czy_palindrom(pomin(s));
3. }
```

4. Sprawdź działanie funkcji z różnymi danymi, w tym z napisami zawierającymi spacje.

### WIEM WIĘCEJ

**Anagram** to wyraz, wyrażenie lub zdanie powstałe przez przestawienie liter albo sylab innego wyrazu, wyrażenia lub zdania. W 1609 r. Galileusz odkrył pierścienie Saturna. Aby zabezpieczyć się przed wścibstwem konkurencji żądnej sławy, włoski astronom zanotował informację o swojej obserwacji za pomocą anagramu: SMAISMRMILMEPOETALEVMIBUNENUGTTAVIRAS. Oznaczało to: *Altissimum planetam tergeminum observavi* (Najwyższą planetę obserwowałem jako potrójną).

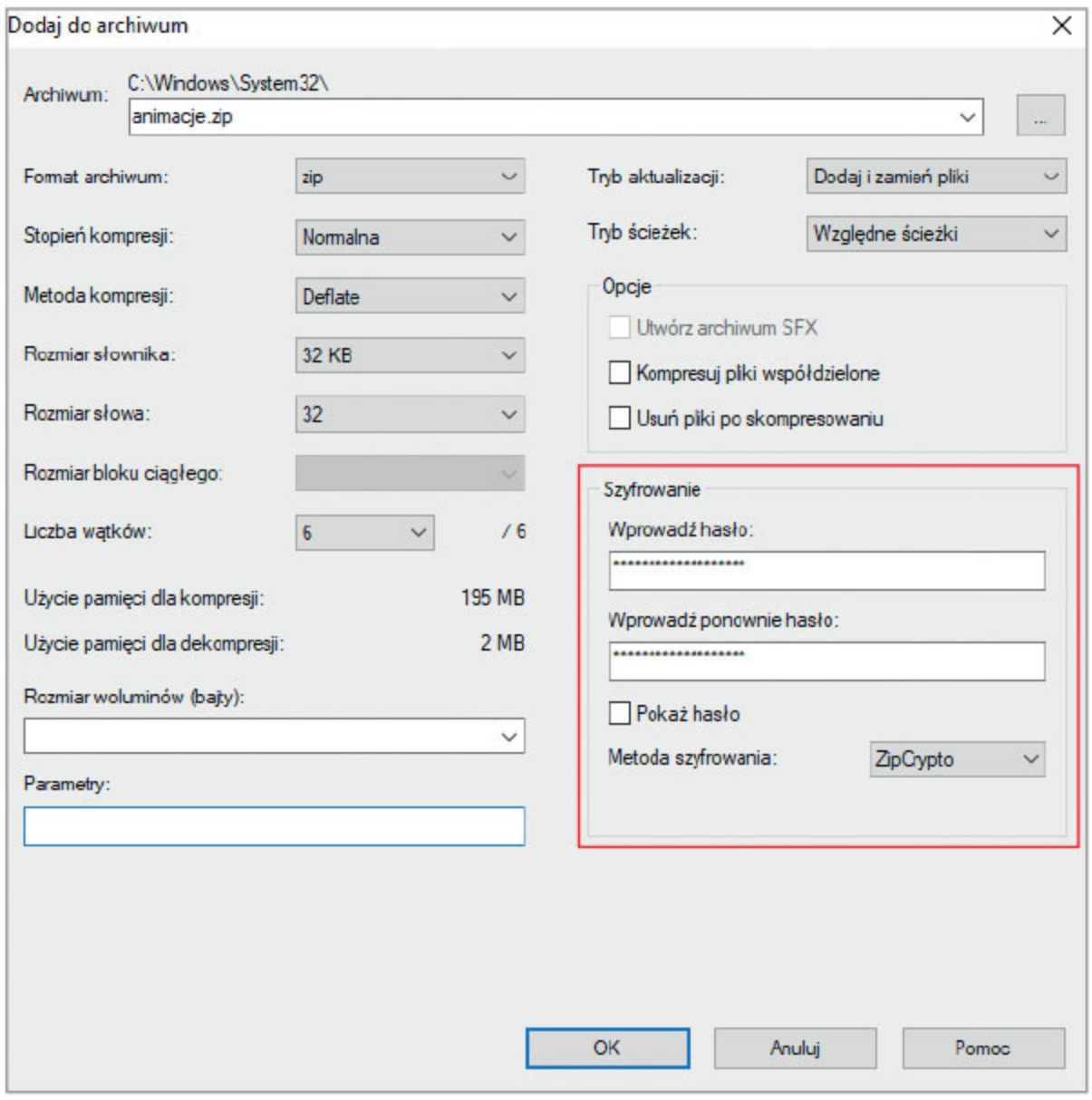


## PRAKTYCZNE WYKORZYSTANIE SZYFROWANIA

Zabezpieczanie danych przez szyfrowanie to jedno z większych wyzwań w społeczeństwie cyfrowym. Szyfrować można m.in. pojedyncze pliki, foldery, dyski lub ich partycje, inne nośniki danych, e-maile albo transmisje danych. Możliwość szyfrowania oferują liczne aplikacje, a także popularne programy do kompresowania danych, takie jak 7-Zip czy WinRAR. Dzięki wykorzystaniu metod informatycznych wiele szyfrów można złamać, ale i tak odczytanie zaszyfrowanej wiadomości może sprawić trudność przeciętnemu człowiekowi.

**POSZUKAJ W INTERNECIE**

Wejdź na stronę **https://pl.khanacademy.org**, w dziale **Informatyka** wybierz *Podróż w krainę kryptografii* i dowiedz się, w jaki sposób ludzie chronili informacje kiedyś i jak robią to dziś.




**Rys. 3.** Szyfrowanie pliku archiwum

## PODSUMOWANIE

- Dostęp do pojedynczego znaku napisu daje indeks, gdzie pierwszy znak ma numer **`0`**, drugi **`1`**, a ostatni **`(długość napisu - 1)`**.
- Szyfrogram to tekst, który został zaszyfrowany. Tekst jawny to tekst, który nie został zaszyfrowany lub został odszyfrowany.
- Szyfry przestawieniowe polegają na zamianie kolejności liter, długość napisu wówczas się nie zmienia.

- Szyfrowanie parkanem polega na zapisaniu liter tekstu jawnego w wierszach. Po usunięciu spacji należy w pierwszym wierszu zapisać pierwszą literę, w drugim drugą, w trzecim trzecią itd., a następnie przejść do kolejnych rzędów i na koniec spisać litery z kolejnych wierszy jako poszczególne słowa. Osoba chcąca odszyfrować tekst musi odczytać najpierw pierwszą literę pierwszego słowa, potem pierwszą literę drugiego słowa itd.
- Palindrom to słowo lub zdanie identyczne niezależnie od tego, czy jest czytane od lewej do prawej, czy od prawej do lewej. Sprawdzamy, czy napis jest palindromem, przez porównanie kolejno par liter: pierwszej i ostatniej, drugiej i przedostatniej itd. Jeżeli napotkamy różne znaki, to już wiemy, że napis nie jest palindromem.
- Zabezpieczanie danych przez szyfrowanie jest nieodzowne w społeczeństwie cyfrowym. Szyfrować można m.in. pojedyncze pliki, foldery, partycje dysków i nośników, e-maile oraz transmisje danych.

## PYTANIA SPRAWDZAJĄCE

1. Jak wypisać pierwszy znak danego napisu? A jak – ostatni?
2. W jaki sposób można odwrócić napis?
3. Omówiony w ćwiczeniu 3 klasyczny sposób sprawdzania, czy napis jest palindromem, dla napisu 10-literowego uruchomi pętlę pięć razy. Ile razy zostanie wywołana pętla, gdy napis ma 11 liter?

## PYTANIA QUIZOWE

1. Przeglądanie napisu co 1 znak można zrealizować za pomocą następującej pętli:

a.
```
for (int i = 1; i < w.size(); i += 1)
        cout << w[i] << endl;
```
b.
```
for (int i = 0; i < w.size(); i += 2)
        cout << w[i] << endl;
```
c.
```
for (int i = 0; i < w.size() - 1; i += 1)
        cout << w[i] << endl;
```
d.
```
for (int i = 0; i < w.size(); i += 1)
        cout << w[i] << endl;
```

2. Jaki znak zostanie wypisany dla napisu `w = "szyfrant"` w wyniku działania polecenia `cout << w[w.size() - 8] << endl;`?
   a. `n`
   b. `s`
   c. `z`
   d. `t`

3. Jak należy zacząć jednowierszowy komentarz?
   a. #
   b. ;
   c. //
   d. $

## ZADANIA DODATKOWE

### Zadanie 1. Ukrywanie tekstu

Przeanalizuj dane w tabeli i zdefiniuj funkcję **`ukryj(string s)`**, której parametrem jest napis złożony z małych liter alfabetu łacińskiego, a wynikiem – napis ze wstawioną co drugą literą wybraną losowo.

| Wywołanie funkcji | Przykładowy wynik |
|---|---|
| `ukryj("zima")` | `"zkitmwau"` |
| `ukryj("alamakota")` | `"axldaamaackvostvac"` |

**Wskazówka:** do losowania wykorzystaj funkcje **`rand()`** i **`srand()`**.

### Zadanie 2. ROT13

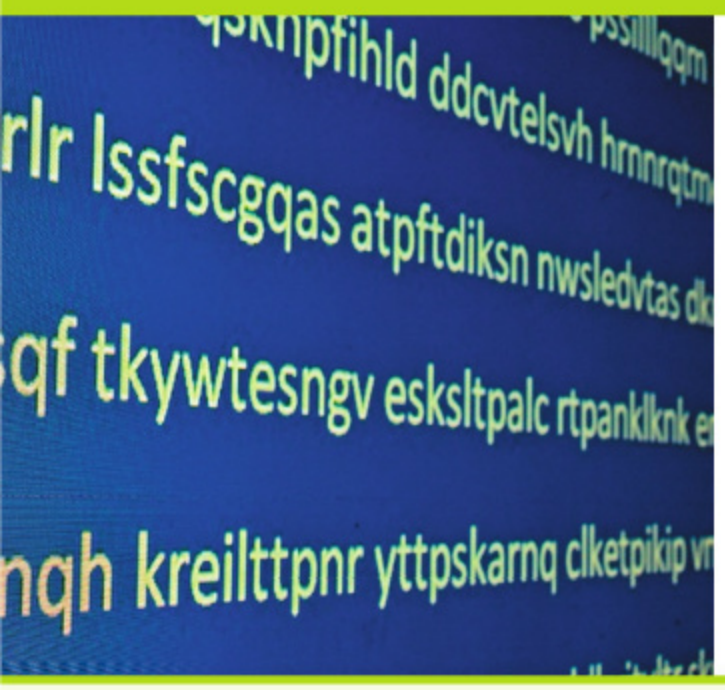
Zapoznaj się z informacją na temat szyfru ROT13 na s. 199, a następnie przeanalizuj dane w tabeli i zdefiniuj funkcję **`rot(string s)`**, której parametrem jest napis zawierający wielkie litery alfabetu łacińskiego, a wynikiem – napis zakodowany według podanej zasady.

| Wywołanie funkcji | Wynik |
|---|---|
| `rot("GRAT")` | `"TENG"` |
| `rot("INFORMACJA")` | `"VASBEZNPWN"` |

### Zadanie 3. Anagram

Przeanalizuj dane w tabeli i zdefiniuj funkcję **`anagram(string s)`**, której parametrem jest tekst jawny składający się z małych liter a, b lub c, a wynikiem napis, w którym wszystkie litery tekstu są ustawione w kolejności alfabetycznej.

| Wywołanie funkcji | Wynik |
|---|---|
| `anagram("acabbca")` | `"aaabbcc"` |
| `anagram("abccbaaaa")` | `"aaaaabbcc"` |



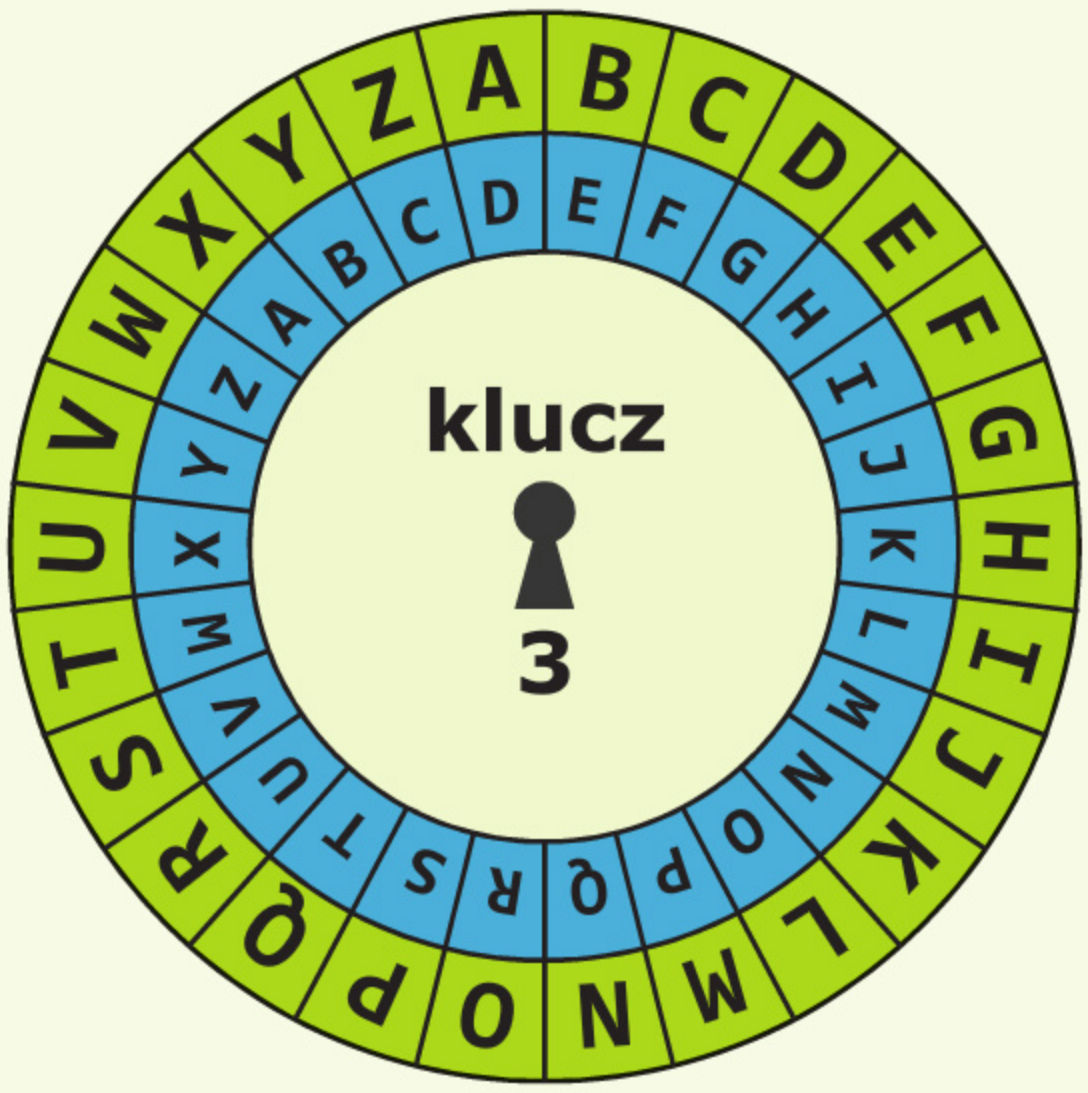

# 18A. Szyfrowanie i deszyfrowanie tekstu

- Podstawowe pojęcia kryptograficzne
- Szyfr Cezara
- Szyfrowanie i odszyfrowywanie z wykorzystaniem kodów ASCII

## ZADANIE NA START

**Odczytaj wiadomość**

Wiesz, że wiadomość **spotkanie o czwartej** zaszyfrowano za pomocą poniższego koła do szyfrowania jako **vsrwndqlh r fczduwhm**. Na tej podstawie odszyfruj wiadomość **vhnuhwqb olvw srg vfkrgdpl**.




## SZYFR CEZARA

W szyfrze podstawieniowym stosowanym przez Juliusza Cezara każda litera tekstu jawnego była zastępowana inną według przyjętego klucza, znanego zarówno odbiorcy, jak i nadawcy wiadomości – i wykorzystywanego do tworzenia szyfrogramu i do jego odszyfrowywania.

**ZAPAMIĘTAJ!**

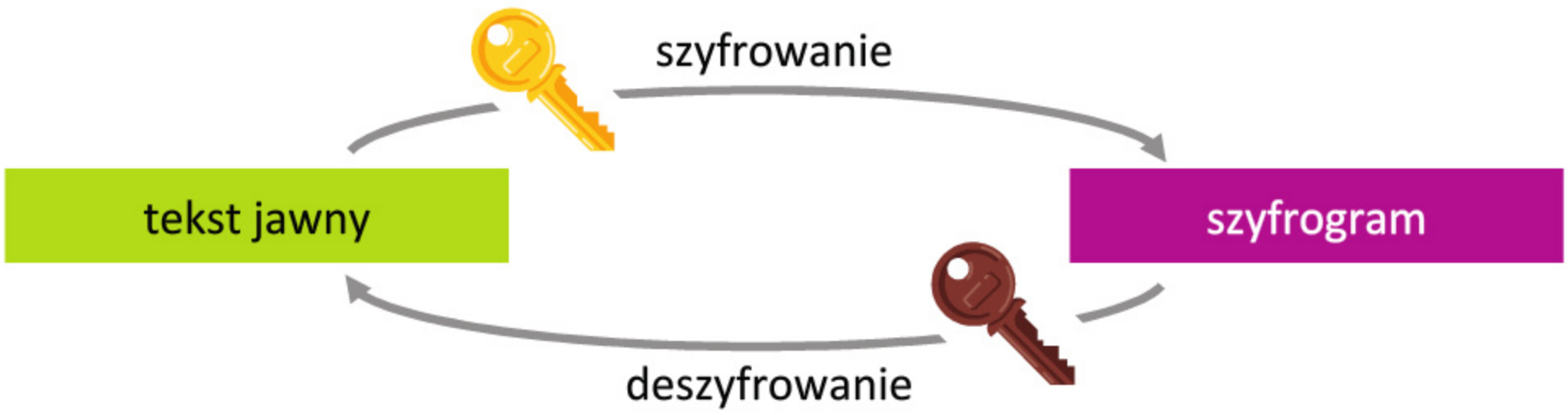
**Klucz** to informacja umożliwiająca wykonywanie pewnej czynności kryptograficznej, np. szyfrowania lub deszyfrowania.

Klucz w szyfrze Cezara to stała wartość, która wskazuje, o ile w prawo litera zaszyfrowana jest oddalona od pierwotnej. Dla danego napisu *t* i klucza *k* każdej literze napisu *t* przyporządkowuje się literę występującą o *k* miejsc dalej w alfabecie. Po dojściu do końca alfabetu należy zacząć liczyć od początku. Klucz 3 oznacza, że literze **a** odpowiada litera **d**, literze **b** – **e**, a literze **z** – **c**. Klucz 7 oznacza natomiast, że literze **a** odpowiada litera **h**, literze **b** – **i**, a literze **z** – **g**. Rzymski wódz i polityk szyfrował listy do swoich przyjaciół za pomocą klucza 3.

Gdy ktoś ma tekst jawny i klucz, może uzyskać szyfrogram. Jedna metoda szyfrowania pozwala na uzyskanie różnych szyfrogramów dla różnych kluczy.

**POSZUKAJ W INTERNECIE**

Sprawdź, na czym polega steganografia.




**Rys. 1.** Proces szyfrowania i deszyfrowania informacji

## WYKORZYSTANIE KODÓW ASCII

Do szyfrowania można użyć kodu ASCII (*American Standard Code for Information Interchange*), który przyporządkowuje każdemu znakowi w komputerze liczbę z zakresu 0–127 (np. kod ASCII litery **a** to 97, litery **k** – 107, litery **z** – 122).

Aby otrzymać kod ASCII danego znaku, należy rzutować typ **char** na typ **int**. Jeśli zastosuje się np. polecenie **int ('a')**, program wypisze liczbę **97**, jeśli będzie to poniższy kod, wypisze kolejno **97**, **98** i **122**.

```
1. cout <<  int ('a') << endl;
2. cout <<  int ('b') << endl;
3. cout <<  int ('z') << endl;
```

**Rys. 2.** Fragment programu generującego kody ASCII dla podanych znaków

Odwrotnie, aby otrzymać znak odpowiadający danemu kodowi ASCII, należy rzutować na typ **char**. Jeśli zastosuje się np. polecenie **char (97)**, program wypisze literę **a**, a jeśli uruchomi się poniższy kod, litery **a**, **b** i **z**.

```
1. cout << char (97) << endl;
2. cout << char (98) << endl;
3. cout << char (122) << endl;
```

**Rys. 3.** Fragment programu przedstawiający znaki na podstawie kodów ASCII

Aby wypisać kolejne znaki odpowiadające kodom ASCII, np. od 33 do 37, należy zastosować pętlę **for**.

```
1. #include <iostream>
2. using namespace std;
3.
4. int main() {
5.     for (int i = 33; i < 38; i++)
6.         cout << i << " " << char (i) << endl;
7.     return 0;
8. }
```

**Rys. 4.** Program umożliwiający wypisanie liczb i odpowiadających im znaków w ASCII

## Ćwiczenie 1. Szyfrowanie znaku

Przeanalizuj dane w tabeli i zdefiniuj funkcję **`szyfruj_znak(char znak, int klucz)`**, której parametrami są – odpowiednio – dowolna mała litera alfabetu łacińskiego oraz liczba całkowita nieujemna mniejsza od 1000, a wynikiem jest litera zaszyfrowana danym kluczem.

| Wywołanie funkcji | Wynik |
|---|---|
| **`szyfruj_znak('a', 3)`** | **`'d'`** |
| **`szyfruj_znak('z', 7)`** | **`'g'`** |

1. Przeanalizuj treść ćwiczenia i podane przykłady.

- Aby otrzymać pozycję danej litery w alfabecie, zamień literę na kod ASCII, a następnie odejmij od niej kod ASCII litery **a** (**97**). Do liczby w ten sposób otrzymanej należy dodać klucz. Następnie trzeba wyznaczyć resztę z dzielenia uzyskanej sumy przez **26** (długość alfabetu łacińskiego). Na koniec do wyniku wystarczy dodać kod ASCII litery **a** (**97**) i zamienić tak uzyskany kod ASCII na literę.




2. Zapisz odpowiednie formuły.

```
int litera_alfabetu = int (znak) - 97
int kod_nowej_litery = (litera_alfabetu + klucz) % 26
char (kod_nowej_litery + 97)
```

3. Zdefiniuj funkcję **szyfruj_znak(char znak, int klucz)**.

```
1. char szyfruj_znak(char znak, int klucz) {
2.    int litera_alfabetu = int (znak) - 97;
3.    int kod_nowej_litery = (litera_alfabetu + klucz) % 26;
4.    return char (kod_nowej_litery + 97);
5. }
```

4. Przetestuj działanie funkcji z różnymi danymi. Jaki klucz trzeba podać, aby zaszyfrować tekst tym samym znakiem, który pojawia się w tekście jawnym?

## Ćwiczenie 2. Szyfrowanie tekstu

Przeanalizuj dane w tabeli i zdefiniuj funkcję **szyfruj(string tekst, int klucz)**, której parametrami są – odpowiednio – ciąg małych liter alfabetu łacińskiego oraz liczba całkowita nieujemna mniejsza od 1000, a wynikiem jest tekst zaszyfrowany danym kluczem.

| Wywołanie funkcji | Wynik |
|---|---|
| **szyfruj("kiedyspotkanie", 13)** | **"xvrqlfcbgxnavr"** |
| **szyfruj("tajne", 3)** | **"wdmqh"** |

1. Przeanalizuj treść ćwiczenia i podane przykłady.

2. Opracuj algorytm.

- Utwórz zmienną pomocniczą **pom**, która będzie stanowić napis pusty, a następnie dopisz do niej każdą kolejną zaszyfrowaną literę tekstu, np. dla **kiedyspotkanie** będą to kolejno litery **x**, **v**, **r**, …, **a**, **v**, **r**. Oznacza to, że wartości zmiennej **pom** to: **"x"**, **"xv"**, **"xvr"**, …, **"xvrqlfcbgxnavr"**. W ten sposób wynikiem funkcji będzie ostatnia wartość zmiennej **pom**.

3. Zapisz funkcję **szyfruj(string tekst, int klucz)**.

```
1. string szyfruj(string tekst, int klucz) {
2.    string pom = "";
3.    for (int i = 0; i < tekst.size(); i++) {
4.       pom = pom + szyfruj_znak(tekst[i], klucz);
5.    }
6.    return pom;
7. }
```

4. Sprawdź działanie funkcji z różnymi danymi. Rozważ, jak za pomocą tej funkcji odszyfrować tekst.

## Ćwiczenie 3. Odszyfrowywanie tekstu

Wynikiem zdefiniowanej poniżej funkcji **deszyfruj(string tekst, int klucz)** powinien być odszyfrowany tekst z danym kluczem.

```
1. string deszyfruj(string tekst, int klucz) {
2.   return szyfruj(tekst, 26 + klucz % 26);
3. }
```

Przeanalizuj dane w tabeli i popraw błąd w funkcji.

| Wywołanie funkcji | Wynik |
|---|---|
| **deszyfruj("xvrqlfcbgxnavr", 13)** | **"kiedyspotkanie"** |
| **deszyfruj("wdmqh", 3)** | **"tajne"** |

1. Przeanalizuj treść ćwiczenia i podane przykłady.

- Odszyfrowanie tekstu to w istocie ponowne zaszyfrowanie z kluczem będącym jego dopełnieniem do 26. Jeśli tekst jawny zaszyfrowano z kluczem 13, to odszyfrowanie polega na zaszyfrowaniu kryptogramu kluczem 26 – 13 = 13 (co oznacza, że należy przesunąć się do przodu o 13 liter).

**WIEM WIĘCEJ**

**Szyfr Cezara** stosunkowo łatwo złamać, czyli odczytać wiadomość bez znajomości klucza. Chociaż wartość klucza jest nieograniczona, to praktycznie istnieje tylko **26 możliwości**. Automatyczna technika łamania polega na analizie częstości liter. Jeżeli tekst jest dostatecznie długi i wiadomo, w jakim napisano go języku, można zbadać częstość występowania liter w szyfrogramie i porównać ją z częstością występowania liter w danym języku. Najlepsze dopasowanie z dużym prawdopodobieństwem wskaże klucz.

## Ćwiczenie 4. Inaczej na pozycjach nieparzystych i parzystych

Przeanalizuj dane w tabeli i zdefiniuj funkcję **szyfruj2(string tekst, int klucz1, int klucz2)**, której parametrami są – odpowiednio – ciąg małych liter alfabetu łacińskiego oraz dwie liczby całkowite, a wynikiem jest tekst zaszyfrowany w ten sposób, że litery na pozycjach parzystych zaszyfrowano kluczem **klucz1**, a na nieparzystych – kluczem **klucz2**.

| Wywołanie funkcji | Wynik |
|---|---|
| **szyfruj2("poufnytekst", 3, 10)** | **"zreixbdhuvd"** |
| **szyfruj2("poczekajnamnie", 123, 5)** | **"uhhsjdfcstrgnx"** |

## ZASTOSOWANIE SZYFROWANIA

Dzięki wykorzystaniu komputera szyfrowanie i deszyfrowanie przebiega szybciej, niż gdyby szyfrowało się informację samodzielnie. Łatwiej jest jednak złamać szyfr komputerowy – chociażby dlatego, że można wówczas wielokrotnie wykonać algorytm odszyfrowywania. W praktyce stosuje się więc bardziej skomplikowane **metody szyfrowania, oparte na kryptografii asymetrycznej**. Zakłada ona istnienie dwóch kluczy: prywatnego, który zna tylko jego właściciel, i publicznego (jawnego), przy czym klucza prywatnego nie da się łatwo odtworzyć na podstawie publicznego. Szyfry asymetryczne wykorzystuje się m.in. w celu zabezpieczenia dostępu do zasobów hasłem lub cyfrowego podpisywania dokumentów.

## PODSUMOWANIE

- W szyfrze Cezara do szyfrowania i deszyfrowania wiadomości stosuje się klucz, który wskazuje, o ile pozycji w alfabecie należy przesunąć literę.
- Klucz to informacja umożliwiająca wykonywanie pewnej czynności kryptograficznej, np. szyfrowania lub deszyfrowania. Gdy ktoś ma tekst jawny i klucz, może uzyskać szyfrogram. Jedna metoda szyfrowania pozwala na uzyskanie różnych szyfrogramów dla różnych kluczy.
- Do szyfrowania tekstów złożonych z liter alfabetu łacińskiego warto wykorzystywać kody ASCII.
- Obecnie w kryptografii stosuje się szyfry asymetryczne oparte na dwóch kluczach – prywatnym i publicznym.

## PYTANIA SPRAWDZAJĄCE

1. Ile różnych kluczy jest w szyfrze Cezara?
2. Jakie wartości kluczy powodują, że tekst jawny i szyfrogram są identyczne?
3. Gdzie obecnie stosuje się elementy kryptografii?

## PYTANIA QUIZOWE

1. Jak nazywa się szyfr, w którym każdą literę tekstu jawnego zastępuje się inną?
   **a.** Podstawieniowy.
   **b.** Przestawieniowy.
   **c.** Symetryczny.
   **d.** Asymetryczny.

2. Ile wynosi kod ASCII litery **`m`**, jeśli kod ASCII litery **`k`** wynosi 107?
   **a.** 108
   **b.** 109
   **c.** 110
   **d.** 111
3. Napis **`tajemnica`** zaszyfrowano szyfrem Cezara i otrzymano **`xeniqrmge`**. Jakiego klucza użyto?
   **a.** 1
   **b.** 2
   **c.** 3
   **d.** 4

## ZADANIA DODATKOWE

### Zadanie 1. Jakim kluczem zaszyfrowano?

Przeanalizuj dane w tabeli i zdefiniuj funkcję **`jaki(string tekst_jawny, string szyfrogram)`**, której parametrami są ciągi małych liter alfabetu łacińskiego, a wynikiem jest klucz z przedziału **`[0, 25]`**, pozwalający zaszyfrować tekst jawny za pomocą metody Cezara, lub wartość **`-1`**, gdy szyfrogram nie mógł powstać z tekstu jawnego w wyniku szyfrowania metodą Cezara.

| Wywołanie funkcji | Wynik |
|---|---|
| `jaki("niespodzianka", "pkgurqfbkcpmc")` | `2` |
| `jaki("zmiana", "anjbxb")` | `-1` |

### Zadanie 2. Złożenie szyfrów

Zaproponuj metodę szyfrowania opartą na złożeniu dwóch szyfrów, czyli na zastosowaniu najpierw jednego szyfru, a potem drugiego – do otrzymanego szyfrogramu. Napisz funkcję szyfrującą i funkcję deszyfrującą.

# KOMUNIKACJA I NAUKA W SIECI

Umiejętności komunikowania się i tworzenia swojego wizerunku w sieci, a także selekcjonowania informacji oraz nauki online nabierają w obecnym, coraz bardziej scyfryzowanym świecie kluczowego znaczenia. Trzeba umieć:

- rozpoznawać wirtualne zagrożenia,
- dostrzegać różnice między przekazem faktograficznym a opinią, oceną i krytyką.

# ABC GRYWALIZACJI

**Użytkownicy szkoleń zapamiętują:**

- **10 % tego, co czytają,**
- **20 % tego, co słyszą,**
- **90 % tego, co robią, nawet jeśli jest to symulacja.**

**80 % osób chciałoby, aby ich nauka bardziej przypominała grę. Grywalizacja to wykorzystanie mechanizmów gier fabularnych i komputerowych w procesie edukacyjnym. Zastosowanie takich technik, jak punkty, poziomy trudności, pasek postępu czy licznik czasu, zwiększa zaangażowanie uczestników kursu. Z kolei pokonywanie kolejnych poziomów i wyzwań czy zdobywanie nagród daje przyjemność i satysfakcję, która zachęca do zwiększenia wysiłków oraz motywuje do dalszej nauki.**

**PUNKTY**




Pozytywne wzmocnienie (nagroda za poprawne wykonanie zadania, miernik sukcesu), kontrola postępów i porównywanie wyników.

**POZIOMY TRUDNOŚCI**



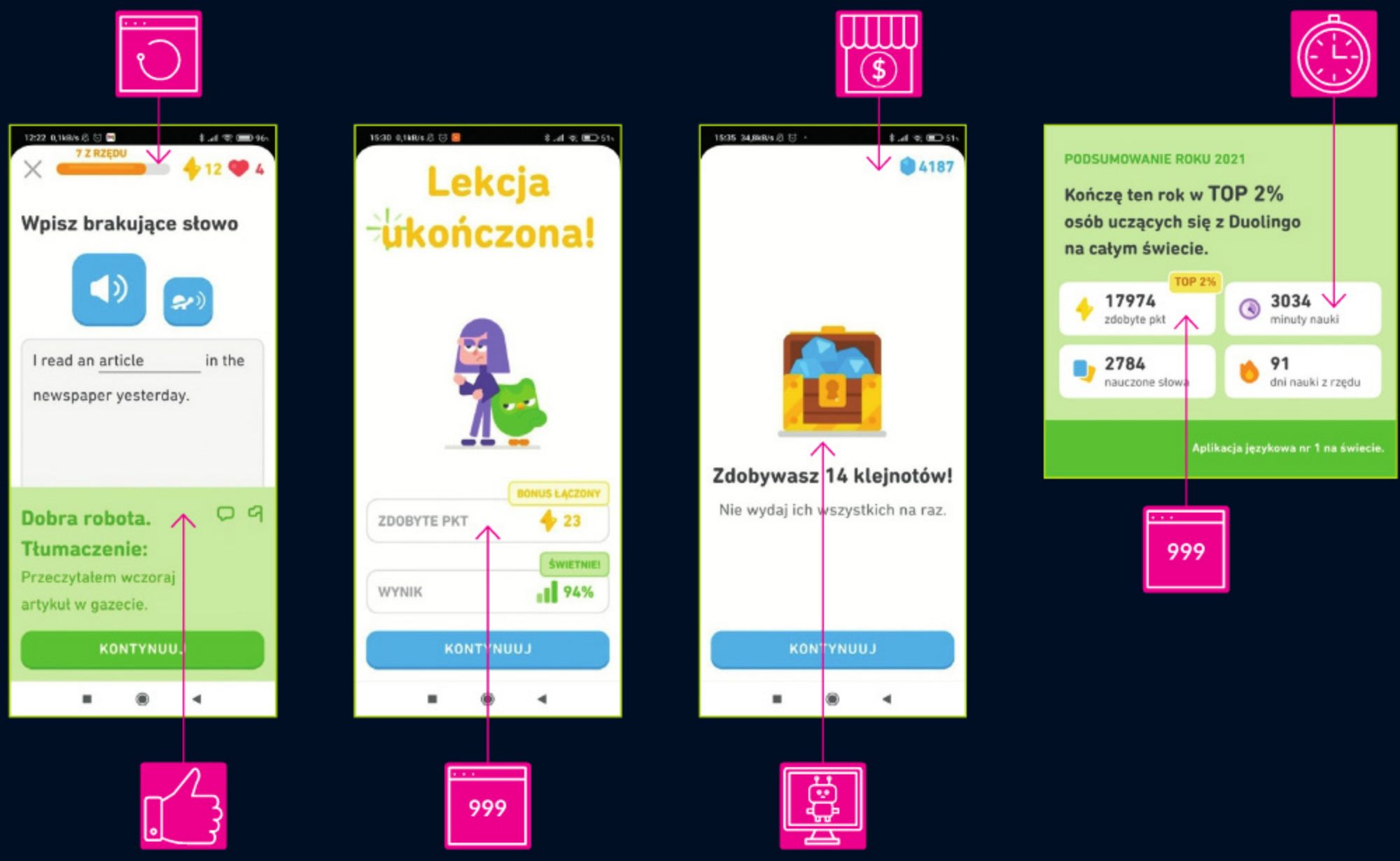
Podział dużego wyzwania na mniejsze, przystępne etapy, umiejętne ustawienie progu wejścia – gra nie może być ani za łatwa, ani za trudna.

**PASEK POSTĘPU**



Informacja dla użytkownika, na jakim znajduje się poziomie, ile zadań już wykonał, jaką partię materiału zrealizował, ile musi zrobić, aby osiągnąć cel.

## GRYWALIZACJA NA DUOLINGO.COM



**RANKING**



Możliwość porównywania się z innymi użytkownikami pod względem liczby zdobytych punktów, wykonanych zadań, pokonanych poziomów itp.

**LICZNIK CZASU**



Mierzenie czasu w rywalizacji z innymi i samym sobą, ocena własnej efektywności.

**WIRTUALNA WALUTA**



Środek płatniczy, który można przeliczyć na nagrody (np. punkty doświadczenia) lub wydać na zakup wirtualnych przedmiotów.

**WIRTUALNE PRZEDMIOTY**



Przedmioty, którymi użytkownik posługuje się w grze.

**WIRTUALNE NAGRODY, ODZNAKI**

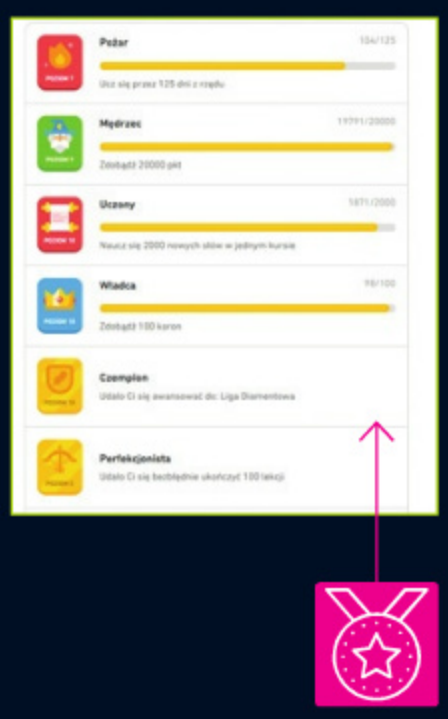


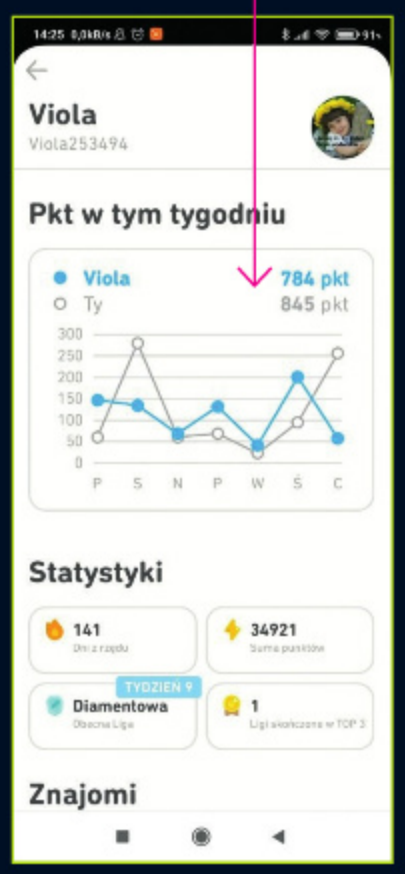
Wizualny sposób wyróżnienia użytkowników, którzy wykonali pewne zadanie lub osiągnęli dany poziom, motywacja do podejmowania kolejnych wyzwań.

**FEEDBACK**



Informacja zwrotna dotycząca danej aktywności, np. pochwała, nagroda, wskazówka, zachęta do podjęcia kolejnej próby.







## PRACA Z INFOGRAFIKĄ

1. Z jakimi formami grywalizacji spotykasz się na co dzień?
2. Jakie działania zwiększają zaangażowanie uczestników kursu?



# 19. Kompetencje medialne i cyfrowe

- Umiejętność wyszukiwania informacji
- Źródła informacji w internecie
- Ocenianie wiarygodności informacji

## ZADANIE NA START

**Ja w sieci**

Prawie każdy ma konto w co najmniej jednym serwisie społecznościowym. Sprawdź, jakie informacje, których nie jesteś autorem, znajdziesz o sobie w sieci.

## KOMPETENCJE MEDIALNE

Świadomy użytkownik mediów sprawnie posługuje się urządzeniami cyfrowymi, umie przeanalizować i krytycznie ocenić zgromadzone informacje, a także wykorzystać zweryfikowane fakty w kontekście swoich potrzeb czy wykonywanego zadania, postępuje racjonalnie, gdy uczestniczy w wirtualnym życiu społecznym i kulturalnym. Innymi słowy: pielęgnuje i rozwija kompetencje medialne.

### Ćwiczenie 1. Kompetencje medialne

Zapoznaj się z infografiką *Czym są kompetencje medialne* zamieszczoną na stronie **https://edukacjamedialna.edu.pl/info/infografiki/**, a następnie przedyskutuj na forum klasy, w jaki sposób kompetencje medialne przyczyniają się do rozwoju aktywności obywatelskiej, umożliwiają pełne uczestnictwo w kulturze i zwiększają szanse na rynku pracy.

**ZAPAMIĘTAJ!**

**Kompetencje medialne** nie są tożsame z umiejętnością obsługi komputera czy tabletu. Osoby mające kompetencje medialne są świadome mechanizmów tworzenia informacji i potrafią bezpiecznie korzystać z mediów, tj. krytycznie oceniać, wybierać i klasyfikować informacje, dostrzegać manipulacje i zagrożenia, rozsądnie zarządzać swoim wizerunkiem.

## ŹRÓDŁA INFORMACJI

Dzięki wartościowym materiałom online można skutecznie poszerzać swoją wiedzę i kompetencje. Źródłem informacji mogą być portale i serwisy informacyjne, lokalne i światowe biblioteki cyfrowe, bazy danych, w tym bazy faktograficzne typu internetowa encyklopedia PWN czy Wikipedia, czasopisma online i ich archiwa, a także platformy e-learningowe, eksperckie blogi i serwisy (prowadzone przez osoby prywatne lub firmy) oraz fora i grupy w portalach społecznościowych.

Platformy e-learningowe oferują materiały dydaktyczne oraz możliwość uczenia się przez aktywne zaangażowanie i rozwijanie twórczych postaw. Oto przykłady:

- Duolingo (**https://pl.duolingo.com**) – platforma do nauki języków obcych, która w języku polskim oferuje kurs języka angielskiego, a po angielsku umożliwia naukę wielu innych języków, m.in. hiszpańskiego, francuskiego, niemieckiego, włoskiego i japońskiego;




**Rys. 1.** Lekcje Duolingo zawierają różne ćwiczenia w formie tekstowej i dźwiękowej, a poza tym wykorzystują system grywalizacji, w ramach którego uczeń otrzymuje nagrody za postępy i rywalizuje z innymi

- OSE IT Szkoła (**https://it-szkola.edu.pl**) – otwarta platforma poszerzająca wiedzę o teorii i zastosowaniach technologii informacyjno-komunikacyjnych, stworzona z myślą o osobach, które chcą podwyższyć swoje kompetencje IT w trybie samokształcenia (poziom materiałów jest zróżnicowany);
- Edukator (**https://edukator.pl**) – zbiór cyfrowych zasobów wraz z narzędziami do tworzenia nowych materiałów, również interaktywnych, a także środowisko do komunikacji i nauczania zdalnego, stworzone z myślą o polskich szkołach.

**WIEM WIĘCEJ**

**MOOC, czyli masowe otwarte kursy internetowe**, organizowane głównie przez uniwersytety, zdobywają coraz większą popularność, a uczestnictwo w nich przyczynia się do podniesienia ogólnego poziomu wiedzy. Niekiedy istnieje też możliwość uzyskania certyfikatu ukończenia takiego kursu. Mimo że są prowadzone w wielu językach, największą popularnością cieszą się kursy anglojęzyczne. Uczestnictwo w takim kursie pozwala niejako przy okazji doskonalić też umiejętności językowe. **Coursea**, **EdX** i **Udacity** to jedne z najbardziej znanych platform oferujących kursy tego typu.

### Ćwiczenie 2. Kurs dla ciebie

Wyszukaj platformy e-learningowe oferujące kursy na wybrany temat, odpowiadające twoim umiejętnościom. Przedstaw krótką charakterystykę kursów i wskaż szkolenie godne polecenia. Wybór uzasadnij z uwzględnieniem przynajmniej dwóch parametrów.

1. Poznaj organizatora kursu.
2. Zwróć uwagę na informacje na temat szkolenia, regulamin i politykę prywatności.
3. Sprawdź zakres i formę kursu.

**Eksperckie blogi i serwisy internetowe** poświęcone danemu zagadnieniu czy określonej branży są prowadzone przez specjalistów w danej dziedzinie. Przedstawiają i komentują nowości, a także w przystępny i rzetelny sposób omawiają typowe i mniej typowe problemy. Przykładowo: blog podróżniczy Duże Podróże (**https://duze-podroze.pl**) podejmuje najróżniejsze tematy dotyczące podróżowania, zawiera wpisy dotyczące konkretnych krajów oraz artykuły związane z planowaniem wyjazdów, pracą w podróży czy fotografią podróżniczą.

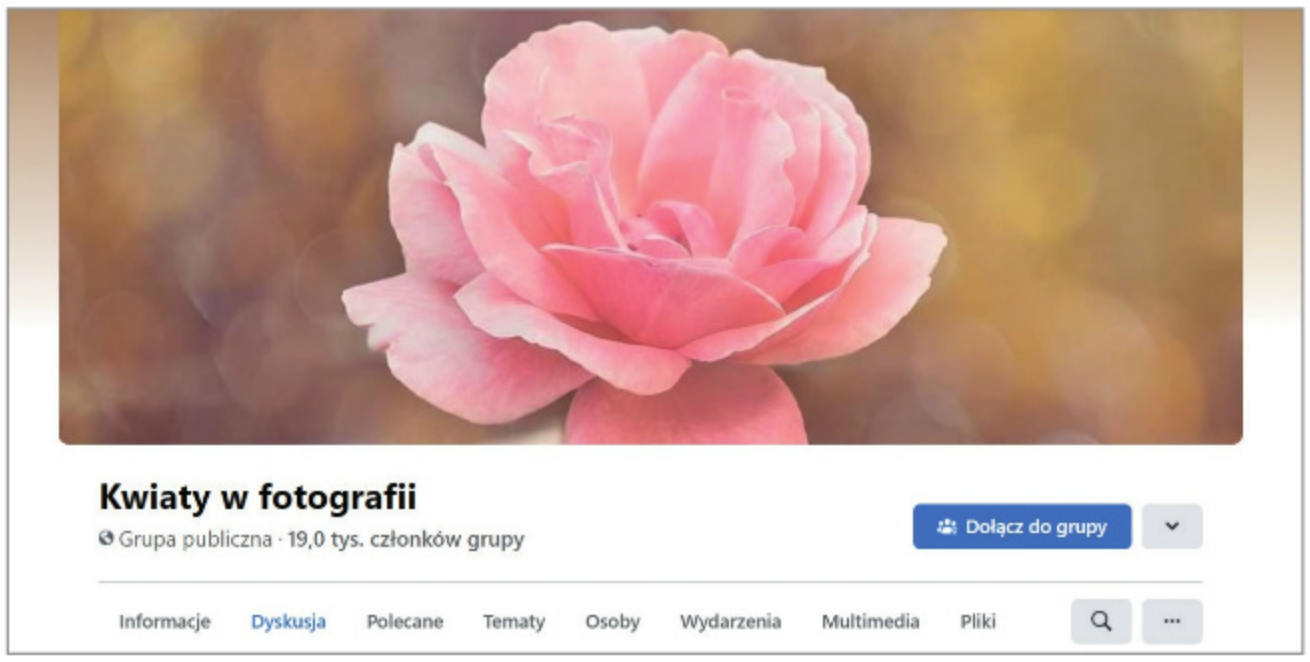
### Ćwiczenie 3. Blog

Podaj przykład blogów eksperckich poświęconych literaturze, muzyce albo gotowaniu.

Fora i grupy w portalach społecznościowych skupiają pasjonatów danej dziedziny – miłośnicy historii wymieniają się swoimi poglądami i materiałami oraz organizują spotkania; zapaleni biegacze wspierają się w treningach i przekazują sobie informacje na temat ważnych wydarzeń, a amatorzy budowania modeli opisują prace nad nowym projektem. Oto przykłady:

- forum dyskusyjne „Strażak" (**https://strazak.pl**) umożliwia wymianę informacji na temat pracy strażaków zawodowych i ochotniczych, w tym taktyki, specjalizacji, sprzętu, rozwoju zawodowego oraz rozwiązywania problemów;
- forum dyskusyjne 4programmers.net (**https://4programmers.net/Forum**) jest przeznaczone dla programistów i umożliwia wymianę informacji na temat poszczególnych języków i technologii, inżynierii oprogramowania czy rozwoju zawodowego;

- „Grupa Miłośników Lotniska Chopina" działa na Facebooku i liczy ponad 12 000 członków, którzy zamieszczają na profilu zdjęcia, filmy i inne materiały dotyczące warszawskiego portu lotniczego;
- „Kwiaty w fotografii" to licząca ponad 19 000 członków grupa działająca na Facebooku; oprócz postów na profilu zamieszczane są własnoręcznie wykonane fotografie kwiatów.




**Rys. 2.** Grupa publiczna na Facebooku „Kwiaty w fotografii"

## Ćwiczenie 4. Forum

Zapoznaj się z zamieszczonym poniżej postem z blogu **devstyle.pl**, a następnie znajdź trzy podobne wypowiedzi osób, które nauczyły się ważnych dla nich rzeczy dzięki aktywności na forum. Postaraj się znaleźć dodatkowe informacje o tych osobach.

> prawie codziennie rano na kilka godzin wchodziłem na wspomniany w poprzednim odcinku portal **CodeGuru**.pl. Forum działało na nim wówczas (lata 2005-2006) niezwykle prężnie. Programistyczne problemy i dylematy wręcz się zeń wysypywały – nie było wtedy StackOverflow. A ja czerpałem z tych problemów pełnymi garściami. Codzienna rutyna obejmowała wejście na forum, wzięcie pierwszego posta z góry i odpowiedzenie na zadane w nim techniczne pytanie. I tak w kółko, ... Czasami pytania były proste i dało się ich pozałatwiać kilkanaście w ciągu jednego posiedzenia. A bywał tak ciężki orzech do zgryzienia, że przez kilka godzin musiałem zgłębiać temat, zanim znajdowałem rozwiązanie. Założenie miałem jedno: nie ma pytań zbyt zaawansowanych, nie ma pytań głupich – biorę wszystko. Jaka (r)ewolucja: **chcę być najlepszy**! Więc muszę umieć pomagać. Musze nauczyć się gromadzić wiedzę i dzielić nią. **Nie mam prawa odpowiedzieć: "nie wiem"**.
> Te poranki dały mi takiego dev-boosta, że przecierałem wirtualne oczy ze zdziwienia. Z tygodnia na tydzień stawałem się zauważalnie lepszy. Obiektywnie zaczynałem być... **ekspertem**. A co najważniejsze: sam zaczynałem się tak czuć, wewnętrznie. "*Try me*". Po kilku miesiącach miałem na swoim koncie tyle odpowiedzi, że pisanie kolejnych szło prawie z automatu. Rozwiązałem tyle problemów, że .NET znałem na wylot. Zmagałem się z prawdziwymi zagwozdkami w prawdziwych projektach z całego kraju. Taki... konsultant właściwie.

Źródło informacji może również stanowić aplikacja o charakterze edukacyjnym. Jednak należy się upewnić, że została ona zaprojektowana i stworzona przez specjalistów w danej dziedzinie, np. bezpłatna aplikacja Anatomy Learning została opracowana przez lekarzy i ekspertów od anatomii, dzięki czemu uczniom, którzy planują studia medyczne, może zastąpić tradycyjny atlas anatomiczny (trójwymiarowe modele organów i kości człowieka przedstawiono w podziale na poszczególne części ciała i układy, a każdy z modeli można dowolnie obracać, przybliżać lub oddalać).

### Ćwiczenie 5. Aplikacja edukacyjna

Pl@ntNet to aplikacja do identyfikacji roślin fotografowanych za pomocą smartfona. Sprawdź, kto jest jej twórcą, jakiego typu informacji dostarcza. Czy zamieszczane tu informacje można uznać za wiarygodne?

## POZYSKIWANIE I OCENA INFORMACJI

Do pozyskiwania informacji warto podchodzić metodycznie, np. z wykorzystaniem poniższego pięcioetapowego schematu.




**Rys. 3.** Schemat pozyskiwania informacji

Gdy ocenia się wiarygodność artykułu znalezionego w sieci, warto uwzględnić siedem poniższych punktów.

1. Jaki był cel napisania tego artykułu?
   - Sprawdź, czy nie masz do czynienia z artykułem sponsorowanym (reklamą) albo żartem.
2. W jakim miejscu opublikowano artykuł?
   - Sprawdź, na jakiej stronie opublikowano artykuł: czy jest to poważny serwis, czy raczej plotkarski, na jakim serwerze się znajduje, kto jest właścicielem i w jakim celu go prowadzi.

3. Przeczytaj treść, aby zrozumieć cały materiał.
   - Zastanów się, czy umiesz oddzielić opinie od faktów – czy artykuł używa wyważonego, precyzyjnego języka, czy też operuje przymiotnikami i słowami nacechowanymi emocjonalnie, straszy, obraża lub wyśmiewa.
4. Czy artykuł nie zawiera błędów językowych, ortograficznych albo edycyjnych?
5. Kim jest autor?
   - Sprawdź, czy autorzy są wiarygodni – czy wypowiada się specjalista z danej dziedziny (poszukaj innych publikacji tego autora), czy wypowiada się poważana instytucja – niektóre instytucje wydają się niezależne, a tymczasem są finansowane przez osoby, w których interesie leży, aby przedstawiać światu ich wersję rzeczywistości. Sprawdź, czy podmiot mający w nazwie określenie „instytut" albo „centrum badań" faktycznie ma taki charakter.
6. Czy podane są wiarygodne źródła?
   - Sprawdź, czy artykuł powołuje się na sprawdzone i aktualne źródła.
7. Czy zdjęcia są wiarygodne?
   - Sprawdź, czy podano źródło zdjęcia (np. nazwisko autora, nazwę agencji prasowej albo pełny adres strony, z której zostało pobrane), czy dużo na nim widać, czy istnieje możliwość, że zostało przerobione w edytorze graficznym, czy szczegóły zgadzają się z treścią artykułu, w jakim kontekście pojawia się na innych stronach.

### Ćwiczenie 6. Ocena artykułu

Wyszukaj w internecie artykuł na temat dokonanego ostatnio odkrycia naukowego i oceń go według powyższych kryteriów, a następnie przedyskutuj swoje spostrzeżenia na forum klasy lub w grupie.

## BEZPIECZNE BUDOWANIE WIZERUNKU W PRZESTRZENI MEDIALNEJ

Aktywne uczestnictwo w społeczności internetowej może zapewnić wiele wymiernych korzyści, ale też stwarzać zagrożenia. Ponieważ wirtualne relacje oznaczają nie tylko komunikację z bliskimi, lecz także nawiązywanie kontaktów między ludźmi zupełnie sobie obcymi, dlatego należy zwracać szczególną uwagę na bezpieczne kształtowanie swojego wizerunku w sieci.

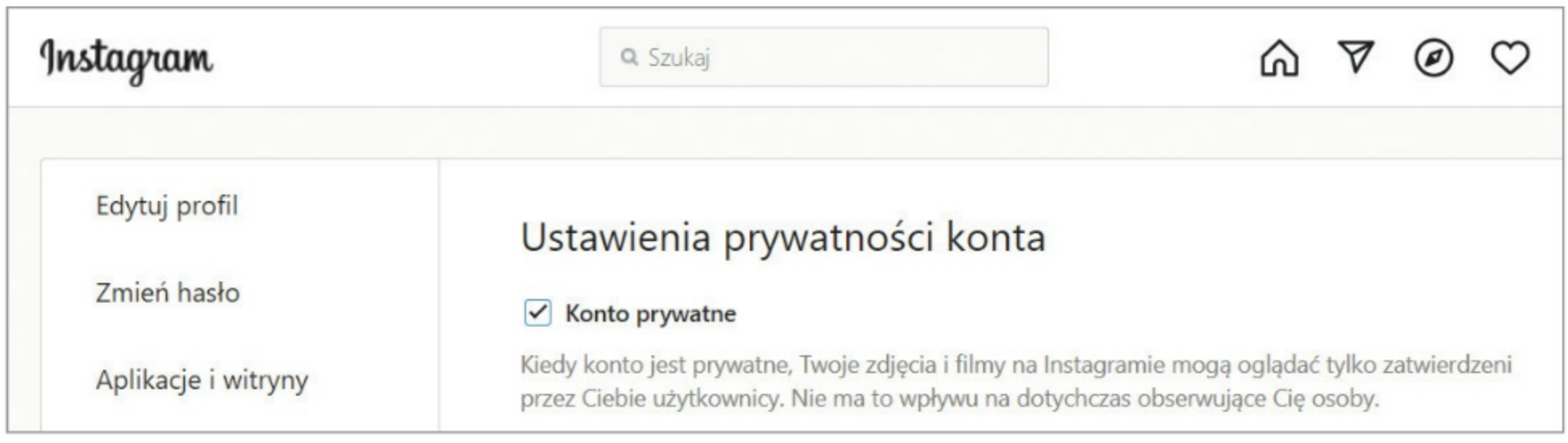
**ZAPAMIĘTAJ!**

**Wizerunek medialny** to obraz danej osoby, firmy lub organizacji, który jest budowany na podstawie tego, co dana osoba (firma, organizacja) prezentuje w mediach społecznościowych na swój temat.

Podczas rejestrowania się w wybranym serwisie należy zapoznać się z polityką prywatności i sprawdzić, w jaki sposób dane przekazywane administratorom portalu będą przetwarzane i chronione. Być może warto nie odpowiadać na pytania dodatkowe o wykształcenie, miejsce pracy czy relacje z innymi użytkownikami albo stworzyć specjalny adres e-mail, który nie będzie wykorzystywany do innych celów. Konto trzeba oczywiście odpowiednio zabezpieczyć za pomocą silnego hasła oraz – jeśli to możliwe – uwierzytelniania dwuskładnikowego.

### Ćwiczenie 7. Na Instagramie

Sprawdź, jakiego typu dane gromadzi o swoich użytkownikach Instagram i w jaki sposób nimi zarządza. Komu przekazywane są zebrane informacje? Do czego są lub mogą być wykorzystywane?




Ważne są również odpowiednie ustawienia – prywatne konto na Instagramie oznacza prywatne treści, publiczne konto oznacza legalną możliwość rozpowszechniania znajdujących się na nim zdjęć czy filmów przez innych użytkowników portalu (z uwzględnieniem przepisów prawa autorskiego i kodeksu cywilnego, zasad opisanych w regulaminie portalu oraz narzędzi udostępnionych przez portal). Jednocześnie trzeba mieć świadomość, że z internetu korzystają osoby, które nie będą przejmować się naruszaniem czyjegoś wizerunku i wykorzystają zdjęcie do stworzenia ośmieszającego mema czy wulgarnego fotomontażu. Należy zawsze myśleć, komu udostępnia się jakie informacje, do czego mogą posłużyć, jeśli wpadną w niepowołane ręce – fałszowanie tożsamości, posługiwanie się czyimś wizerunkiem w celach reklamowych, wykorzystywanie intymnych zdjęć czy filmów do celów pornograficznych, pornografia dziecięca to powszechne problemy związane z obecnością w mediach społecznościowych. A ponieważ w sieci nic nie ginie, a szybko się namnaża, bardzo trudno usunąć krzywdzące treści.

**ZAPAMIĘTAJ!**

Nawet jeśli konto ma charakter prywatny, informacje mogą wyciec – wystarczy, że jeden z zaakceptowanych odbiorców prywatnego profilu bez zgody autora udostępni je na swoim profilu publicznie lub dla szerszego grona osób, które rozpowszechnią treści dalej. Dlatego warto wszystkie zamieszczane treści traktować jako dostępne publicznie, dla każdego.

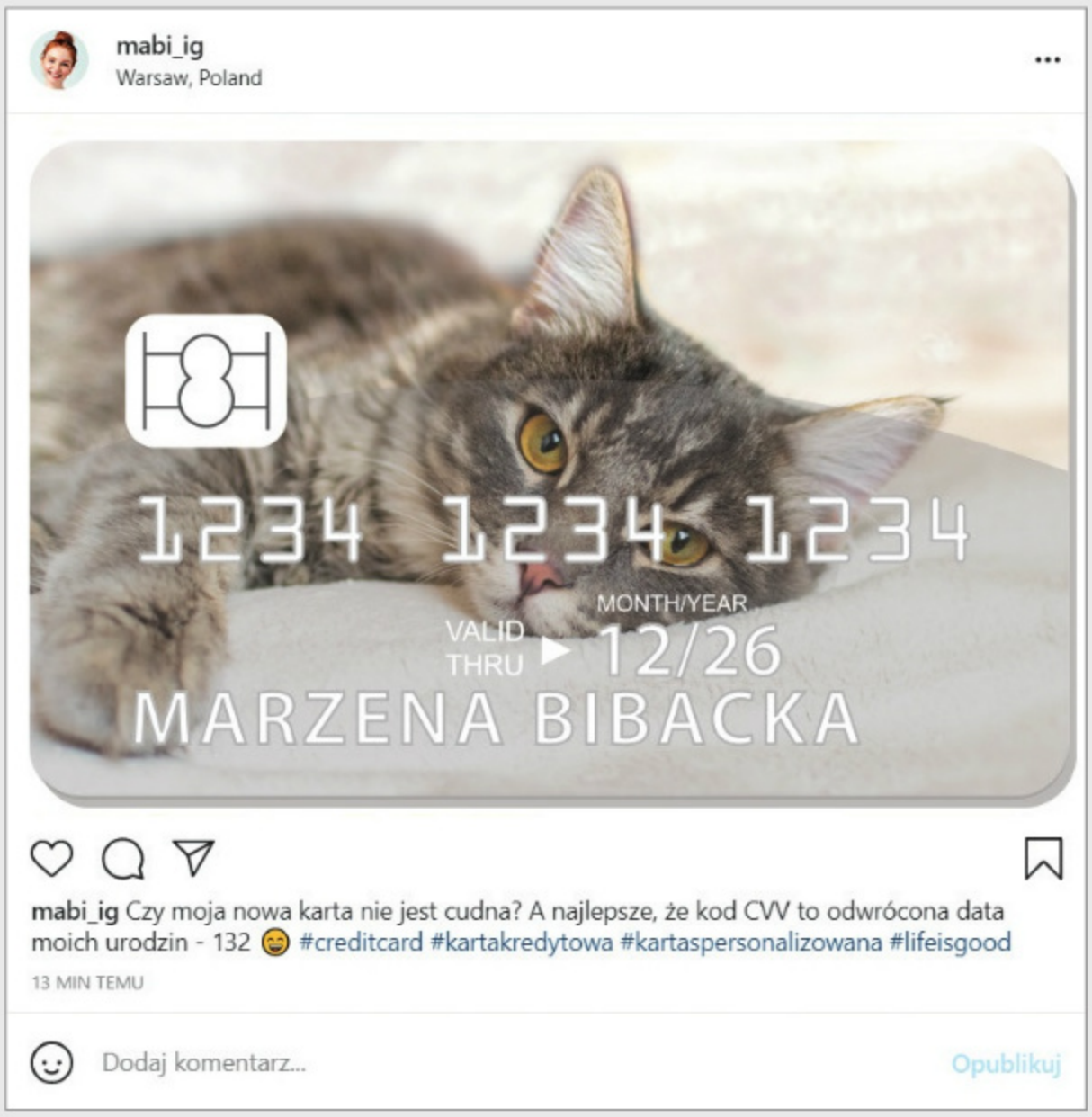
## Ćwiczenie 8. Potencjalne skutki publikacji

Jakie konsekwencje mogą mieć następujące posty zamieszczone w mediach społecznościowych?

**A**




**B**




**C**

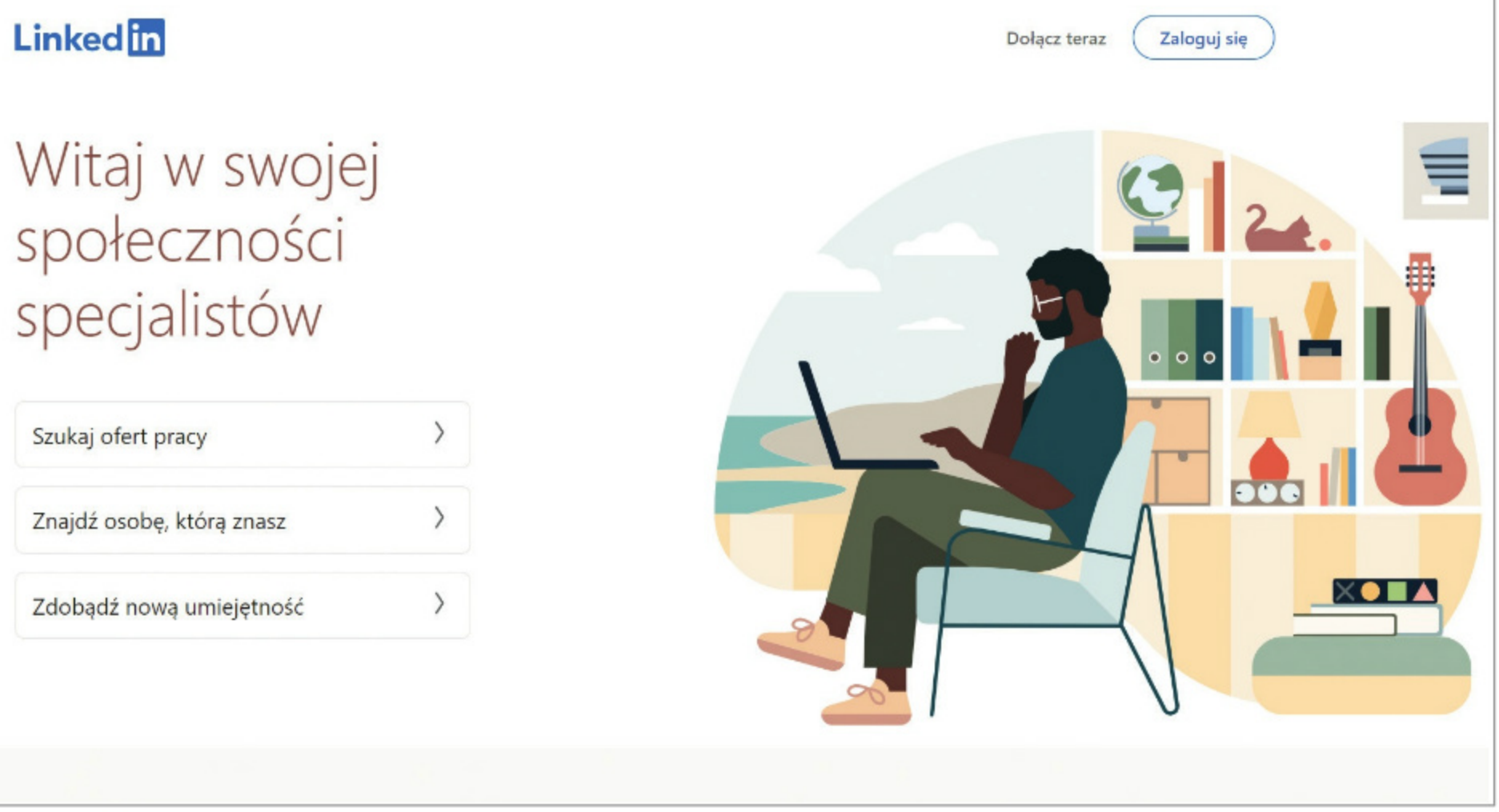


### Ćwiczenie 9. Zastanów się dwa razy, zanim coś opublikujesz

Treści kontrowersyjne w mediach społecznościowych, niezależnie od tego, czy były zamieszczane prywatnie, czy publicznie, mogą w przyszłości dotrzeć do osób, które zainteresują się tym, co działo się jakiś czas temu, np. do potencjalnego pracodawcy. Znajdź w internecie trzy przykłady ukazujące konsekwencje nieodpowiedzialnego zarządzania treścią.

**W PRAKTYCE**

**Serwisy społecznościowe** sprawdzają się również na płaszczyźnie zawodowej. Portale monitorujące rynek pracy, takie jak **GoldenLine** czy **LinkedIn**, umożliwiają poznanie osób o tej samej profesji lub z tej samej branży, tworzenie marki osobistej i (lub) firmowej, budowanie sieci kontaktów, a także aktywne i bierne szukanie pracy (można samodzielnie przeglądać ogłoszenia w zakładce z ofertami albo odpowiednio przygotować profil, tj. zaznaczyć chęć rozwoju kariery oraz uzupełnić słowa kluczowe dla danej branży i czekać na kontakt ze strony pracodawcy bądź rekrutera).




## NETYKIETA

Niestety, interakcje online ze względu na fałszywe poczucie anonimowości otwierają pole różnym nieetycznym zachowaniom. Należy pamiętać, że nie wolno publikować treści rażąco naruszających zasady współżycia społecznego, w tym zawierających słowa powszechnie uznane za niecenzuralne, znieważających lub zniesławiających inną osobę, obrażających inne narodowości, ludzi innych ras, innej orientacji seksualnej, uczucia religijne lub nawołujących do nienawiści na tle różnic narodowościowych, etnicznych, rasowych czy wyznaniowych. Nie wolno spamować, trollować czy łamać praw autorskich. Naruszanie netykiety należy zgłaszać moderatorom lub administratorom serwera, którzy mogą wyciągnąć stosowne konsekwencje – upomnieć czy też ukarać zablokowaniem konta lub usługi na określony czas.

### Ćwiczenie 10. Nie zawsze jest tak, jak być powinno

Zapoznaj się z poniższym postem z forum Militaria, a następnie wybierz forum lub grupę w portalu społecznościowym i znajdź przykład wątku, w którym dochodzi do przekroczenia netykiety oraz do upomnienia przez inną osobę. Udokumentuj wypowiedź – wykonaj zrzut ekranu z istotnymi fragmentami interakcji.

> ★ Netykieta na forum Militaria
>
> [redacted] 12.04.14, 15:21
>
> Zauważyłem, że zaczynają się skargi na cenzurę na forum. Zatem pragnę przypomnieć skarżącym, że na forach obowiązuje pewien regulamin dyskusji oparty na **Netykiecie** pl.wikipedia.org/wiki/Netykieta
> Mimo że forum Militaria jest specyficznym forum i tolerowany tu jest pewien specyficzny męski styl bycia, jednak nie powinno się przeginać w tym stylu.
> Mówiąc krótko: trzymajmy się pewnych norm, a nikt nie będzie wycinany, łącznie ze mną:-)
> Pozdrawiam.

## PODSUMOWANIE

- Kompetencje medialne pozwalają na świadome, skuteczne korzystanie z technologii komunikacyjno-informacyjnych. Pozwalają selekcjonować informacje w kontekście potrzeb informacyjnych i wykonywanego zadania.
- Źródłem informacji w internecie są m.in. portale i serwisy informacyjne, lokalne i światowe biblioteki cyfrowe, lokalne i światowe bazy danych, czasopisma online i ich archiwa, platformy e-learningowe, eksperckie blogi i serwisy, grupy w portalach społecznościowych.
- Kontakty online otwierają pole nieetycznym zachowaniom. Każde rażące naruszenie netykiety powinno się zgłaszać moderatorom lub administratorom serwera.
- Do pozyskiwania informacji najlepiej podchodzić systematycznie, tj. określić problem, wybrać strategię wyszukiwania informacji, wyszukać dostępne informacje, a następnie ocenić je pod względem wiarygodności i rzetelności.
- Bezpieczne budowanie wizerunku w przestrzeni medialnej wymaga odpowiednich ustawień kont w serwisach, przestrzegania zasad dotyczących komunikacji w sieci oraz zdrowego rozsądku. Nawet najlepsze hasła, ustawienia i antywirusy nie pomogą, jeśli właściciel profilu na Facebooku, YouTubie czy Instagramie nie będzie sam chronić swojej prywatności.

## PYTANIA SPRAWDZAJĄCE

1. Jakie są kolejne kroki wyszukiwania i analizowania informacji?
2. W jakiego typu źródłach online można szukać materiałów do samokształcenia?
3. Jakie są główne kryteria oceny rzetelności informacji?

## PYTANIA QUIZOWE

1. Których umiejętności nie zalicza się do kompetencji medialnych?
   **a.** Sprawnego posługiwania się urządzeniami cyfrowymi.
   **b.** Selekcji i analizy informacji.
   **c.** Twórczego wykorzystania informacji.
   **d.** Szybkiego czytania tekstów.
2. Dlaczego przy wyszukiwaniu informacji warto notować źródła? Wskaż stwierdzenie nieprawdziwe.
   **a.** Podaną informację należy opatrzyć autorstwem.
   **b.** Pomaga to przy ocenie wiarygodności.
   **c.** Można potem wrócić do źródła informacji i zbadać kontekst.
   **d.** Bez zapisanego źródła plik zniknie z naszego komputera.
3. Czym jest wizerunek medialny?
   **a.** Jest to zdjęcie profilowe użytkownika w serwisie społecznościowym.
   **b.** Jest to opis zainteresowań użytkownika serwisu społecznościowego.
   **c.** Jest to obraz budowany na podstawie tego, co dana osoba prezentuje w mediach społecznościowych na swój temat.
   **d.** Jest to album zdjęć użytkownika serwisu społecznościowego.

## ZADANIA DODATKOWE

### Zadanie 1. Blog przedmiotowy

Zastanów się, czy prowadzenie blogu na temat wykorzystania technologii informacyjno-komunikacyjnych w nauce ulubionego przedmiotu, np. zastosowań arkusza kalkulacyjnego w fizyce, może pomóc jego autorowi w nauce fizyki. Czy regularne zajmowanie się blogiem to odciąganie od nauki, czy raczej zachęta do dalszego rozwoju? Uzasadnij swoje odpowiedzi.

### Zadanie 2. Szkolenie przez komunikator

Zaplanujcie i przeprowadźcie w parach krótkie szkolenie na dowolny temat w formie wideokonferencji. Wykorzystajcie współdzielenie ekranu, które pozwala na przekazywanie materiałów pisanych i wizualnych. Do oceny przedstawcie projekt, nagranie ze szkolenia oraz wnioski po jego realizacji.

### Zadanie 3. Nauka języków obcych przez internet

Zaproponuj 10 sposobów, dzięki którym możesz doskonalić swoje umiejętności posługiwania się językiem obcym z wykorzystaniem internetu.



# 20. E-learning

- Wady i zalety nauki przez internet
- Zasady pracy na platformie e-learningowej
- Aktywny udział w szkoleniu

## ZADANIE NA START

**Organizacja nauki**

Każdy ma własne upodobania w kwestii czasu nauki. Jedni uczą się z samego rana, inni wolą pracować wieczorem, czasem do późna. Zapytaj o preferencje kolegów i koleżanki w klasie. Czy jest wśród was więcej skowronków, czy sów? Wyniki przedstaw w tabeli i na wykresie.

## E-LEARNING

Podstawowe zadania platformy e-learningowej to:

- udostępnianie materiałów,
- administrowanie użytkownikami,
- sprawdzanie wiedzy,
- informowanie o postępach uczniów,
- analizowanie procesu nauczania i uczenia się oraz raportowanie.

Dobrze przygotowane szkolenie online ma starannie zaplanowaną ścieżkę aktywności uczącego się, a jednocześnie pozwala na indywidualizację procesu nauki poprzez dostosowanie zakresu, intensywności oraz poziomu kursu do możliwości danej osoby. Uczestnicy mają dostęp do wykładów pisemnych oraz nagranych, do których można wracać wielokrotnie, prezentacji multimedialnych, animacji, studiów przypadku, a także ćwiczeń, testów oraz zadań otwartych, które przekazują do sprawdzenia i opisania nauczycielowi (mentorowi). Należy zauważyć, że w klasie, w której nauczyciel i uczniowie spotykają się w tym samym miejscu i czasie, oprócz przekazu formalnego, związanego z normalnym przebiegiem zajęć, pozostaje wiele miejsca na komunikację nieformalną. W przypadku szkoleń online możliwość interakcji z nauczycielem oraz innymi uczestnikami szkolenia jest dość ograniczona. Uczestnicy mogą odbywać indywidualne konsultacje na czacie lub podczas wideokonferencji, a z kolegami komunikują się czasem niewerbalnie na forum.

**POSZUKAJ W INTERNECIE**

Zapoznaj się z historią e-learningu – sprawdź, kiedy powstały pierwsze szkolenia tego typu.

Od uczącego się kształcenie zdalne wymaga inicjatywy oraz aktywności. Istotny problem mogą stanowić samodyscyplina i systematyczność. Częstym argumentem przy zapisywaniu się na szkolenia online jest spostrzeżenie, że można uczyć się zawsze i wszędzie – wystarczy dostęp do internetu. Łatwiej jest, gdy ktoś mobilizuje uczestnika szkolenia do pracy, trudniej, gdy trzeba wykazać się wewnętrzną motywacją – a tak jest zazwyczaj przy nauce przez internet. Czy większość osób nie odkłada nauki na ostatnią chwilę? Poza tym zdarza się, że uczestnik e-kursu poprzestaje na samym otwarciu platformy, zamiast skupiać się na rzeczywistym zrozumieniu i przyswojeniu materiału.

## AKADEMIA KHANA JAKO PRZYKŁAD PLATFORMY E-LEARNINGOWEJ

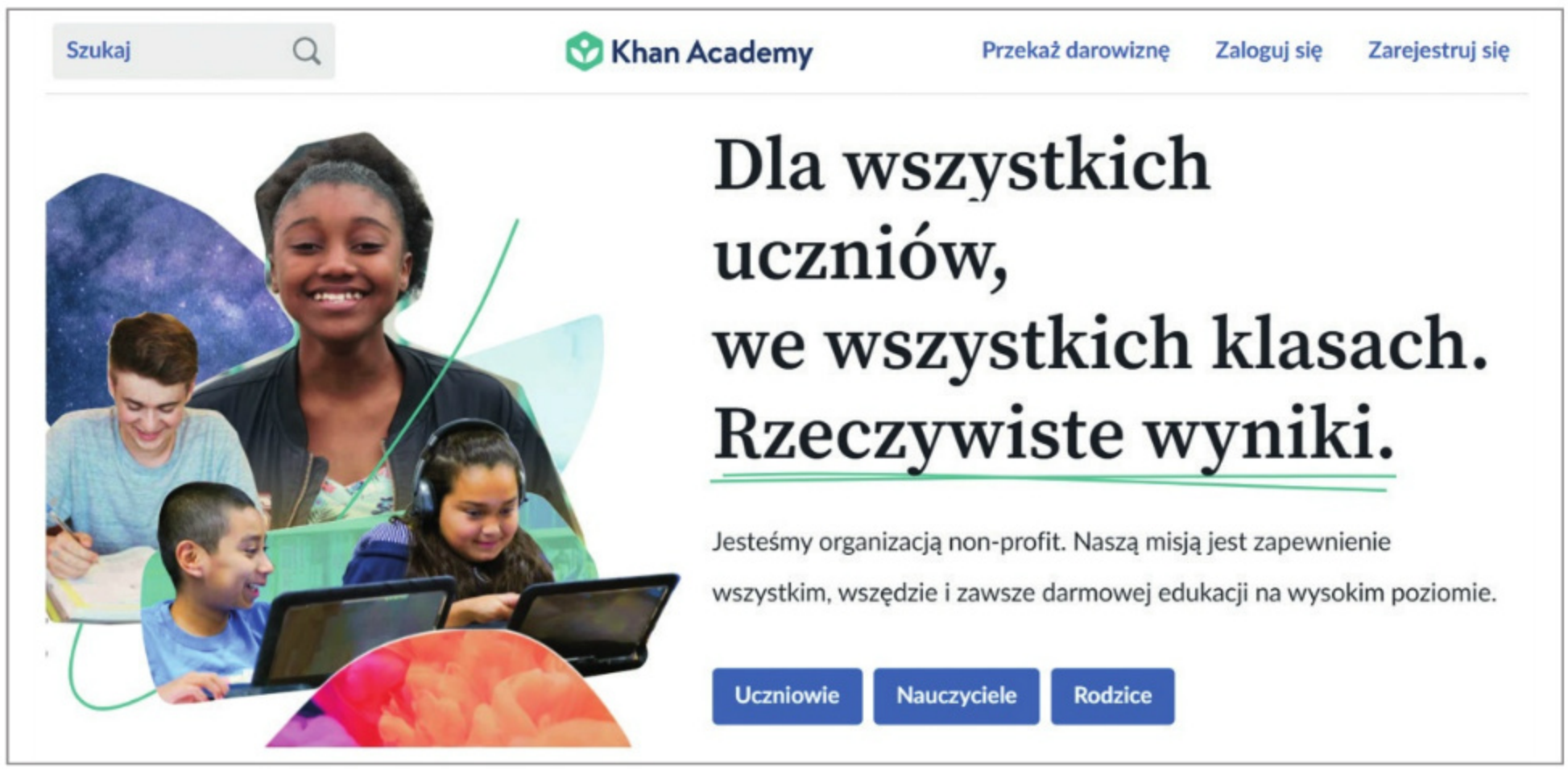
Akademia Khana (**https://pl.khanacademy.org**) to organizacja edukacyjna, której misją jest zapewnienie światowej klasy bezpłatnej edukacji dla wszystkich i wszędzie. Oferuje materiały dydaktyczne z nauk ścisłych, ekonomii i finansów, nauk humanistycznych i sztuki oraz informatyki, w tym około 9000 wykładów nagranych w postaci filmów umieszczonych w serwisie YouTube oraz panel indywidualnych planów nauczania, który daje możliwość pracy we własnym tempie, w szkole lub poza nią. W celu zgromadzenia wyspecjalizowanych treści organizacja współpracuje z poważanymi amerykańskimi instytucjami, np. NASA czy The Museum of Modern Art, oraz uczelniami, np. California Academy of Sciences i MIT.

**POSZUKAJ W INTERNECIE**

Dowiedz się, co było bodźcem do utworzenia Akademii Khana.

## ZAKŁADANIE KONTA W AKADEMII KHANA

Aby wziąć udział w szkoleniu, najpierw należy się zarejestrować. Użytkownik, który ma konto, może nie tylko obejrzeć publikowane filmy i rozwiązać zadania, lecz także skorzystać ze wsparcia w nauce, badać swoje mocne i słabe strony, śledzić postępy.



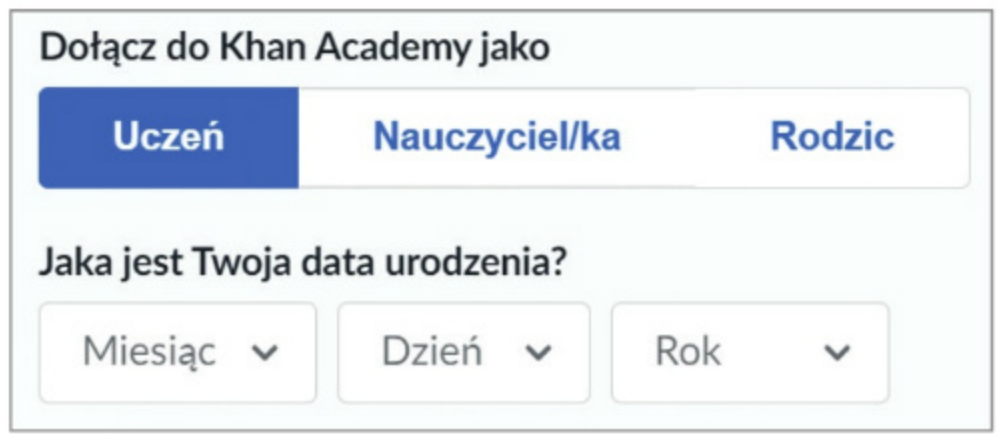

**Rys. 1.** Fragment strony głównej Akademii Khana

## Ćwiczenie 1. Zakładanie konta

Jeśli nie masz jeszcze konta na platformie Akademii Khana, zarejestruj się jako uczeń i uzupełnij profil użytkownika. Zapamiętaj login i hasło, gdyż będą ci potrzebne do wykonania kolejnych ćwiczeń.

1. Zarejestruj się. Kliknij przycisk **Zarejestruj się**, aby przejść do formularza rejestracji.

2. Wybierz opcję **Uczeń**, a następnie podaj datę urodzenia, proponowaną nazwę użytkownika, hasło i e-mail jednego z rodziców lub opiekunów prawnych (rodzic lub opiekun musi potwierdzić, że zgadza się na założenie przez ciebie konta w Akademii Khana). Twoja nazwa użytkownika pojawi się w adresie Akademii Khana.




- Jeśli masz konto Google lub profil na Facebooku, możesz się zarejestrować za ich pośrednictwem.




3. Uzupełnij profil użytkownika – wybierz nazwę użytkownika (pojawi się w adresie Akademii Khana), awatar i tło; wszystko, co zrobisz jako zalogowany użytkownik, zostanie zapisane w profilu.

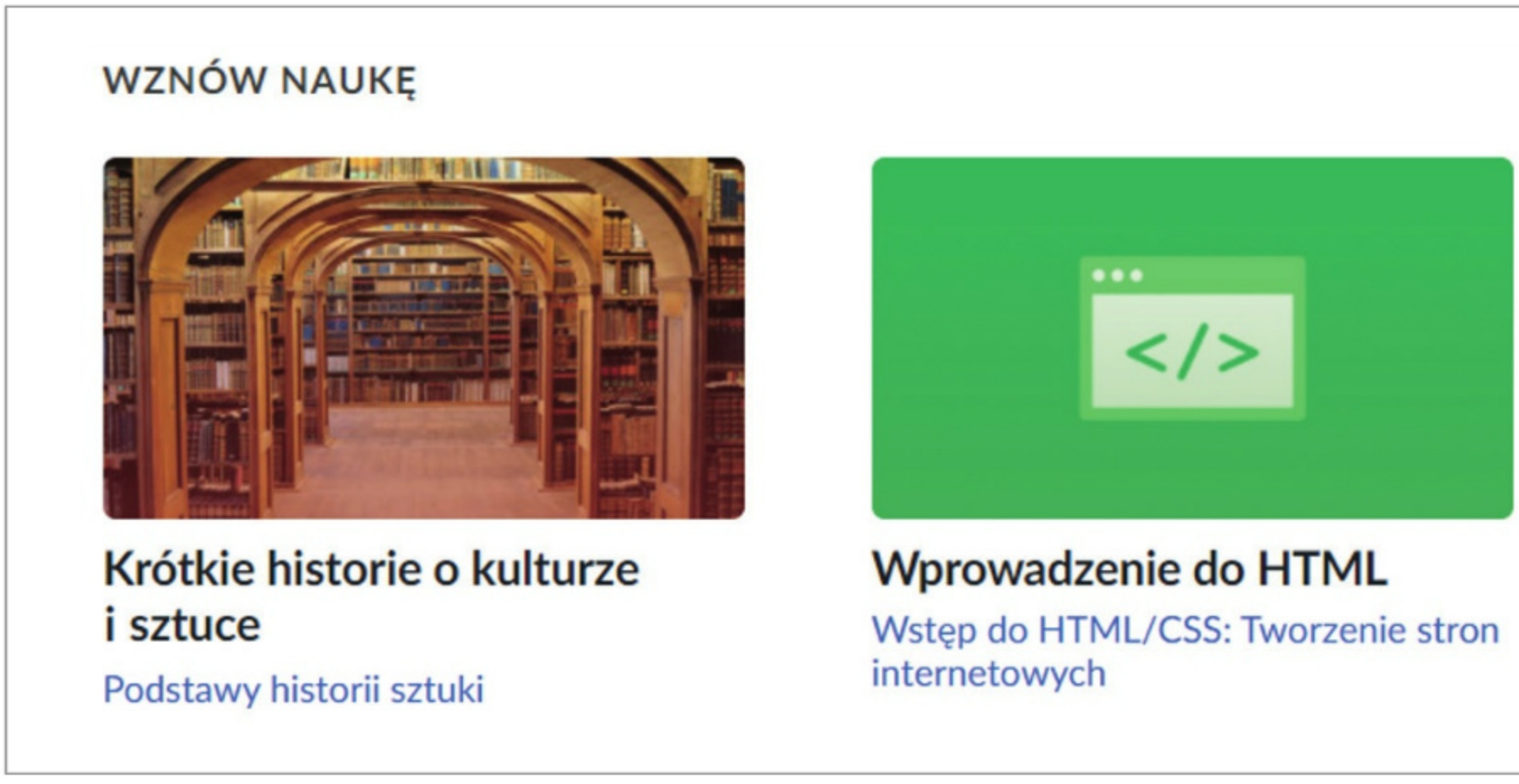
### NAUKA W AKADEMII KHANA

Udział w szkoleniach Akademii Khana przypomina pracę z samouczkiem. Po pierwsze, nie obowiązują tu żadne terminy – można podejmować aktywność szkoleniową, kiedy się chce i jak długo się chce. Po drugie, rola nauczyciela sprowadza się do obserwacji postępów. Po trzecie, skoro każdy pracuje w swoim tempie, interakcja między uczącymi się jest utrudniona.

Na platformie zdobywa się odznaki za realizację kolejnych aktywności, można też obserwować, jak pracują inni, np. gdy publikują swoje prace.

Portal zazwyczaj proponuje uczestnikowi quiz sprawdzający na start, aby spersonalizować ścieżkę nauki. Uczniowie mogą badać swoje mocne i słabe strony, obserwować postępy i czerpać motywację do dalszej pracy.

W każdej chwili możesz wrócić do rozpoczętego szkolenia i kontynuować naukę. Wystarczy wybrać odpowiednią ikonę.




**Rys. 2.** Wznawianie nauki

### Ćwiczenie 2. Uczestnictwo w kursie

Zaloguj się do swojego konta na platformie Akademii Khana, a następnie wybierz kurs **Wstęp do HTML/CSS: Tworzenie stron internetowych**. Weź aktywny udział w tym szkoleniu – zdobyte umiejętności wykorzystasz podczas realizacji rozdziału *Interaktywne strony WWW*. Zaplanuj czas na samodzielną naukę.

1. Wybierz szkolenie.

   **Przedmioty → Informatyka → Programowanie → Wstęp do HTML/CSS: Tworzenie stron internetowych**

2. Jeśli nie znasz podstaw HTML-a, zacznij od *Wprowadzenia do HTML* – dowiesz się, czym jest HTML oraz jak stworzyć stronę internetową z tekstem i obrazkami. Jeśli znasz podstawy tego języka, przejdź do *Wprowadzenia do CSS* – dowiesz się, jak stosować kaskadowe arkusze stylów, aby działały wybiórczo na podstawie elementu, klasy czy ID oraz zmieniały kolor strony. Jeśli te zagadnienia również są ci znane, wybierz następny temat.

3. Realizuj kolejne aktywności i zdobywaj punkty.

4. W razie potrzeby skorzystaj ze wskazówek **Co to jest?**, które mają pomóc w rozwiązywaniu problemu.

5. Zapisz **Spin-off** rozwiązywanego projektu. Zobaczysz go na swojej liście **Moje Projekty**. Łatwo będzie ci go odnaleźć.

### Ćwiczenie 3. Jak radzić sobie z trudnościami?

Zastanów się nad tematem zdobywania wiedzy przez internet. Krótko scharakteryzuj wady i zalety związane z takim rodzajem nauki. Weź pod uwagę predyspozycje uczącego się. Opisz trzy problemy związane z nauką przez internet i zaproponuj konkretne wskazówki, jak sobie z nimi poradzić.

**WIEM WIĘCEJ**

Do nauki w Akademii Khana można wykorzystać oprócz komputera aplikację zainstalowaną na tablecie lub smartfonie. Ustawienia konta synchronizują się z **khanacademy.org**, więc są zawsze aktualne. Do nauki nie trzeba mieć połączenia z internetem, można uczyć się w trybie offline, tylko wcześniej należy pobrać niezbędną zawartość.

## PODSUMOWANIE

- Samokształcenie z wykorzystaniem internetu pomaga rozwijać zainteresowania, uzyskać pomoc w nauce lub nadrabiać zaległości. Każdy może ćwiczyć w swoim tempie, powtarzać trudniejsze fragmenty, wykonywać ćwiczenia pod okiem wirtualnego nauczyciela, który nie pozwoli opuścić żadnego elementu ćwiczenia.
- Uczestnictwo w kursie e-learningowym wymaga solidności, systematyczności i podtrzymywania motywacji. Aby ukończyć taki kurs, trzeba zazwyczaj spełnić określone kryteria, np. uzyskać wymaganą liczbę punktów. Przy okazji uczestnik kursu doskonali swoje umiejętności związane z posługiwaniem się technologiami informacyjno-komunikacyjnymi (TIK).

## PYTANIA SPRAWDZAJĄCE

1. Jak wygląda organizacja kursu online?
2. Jak zorganizować sobie naukę na kursie e-learningowym z uwzględnieniem własnych możliwości i preferencji? Gdzie szukać wsparcia?
3. Jak można wykorzystać Akademię Khana do własnego rozwoju?

## PYTANIA QUIZOWE

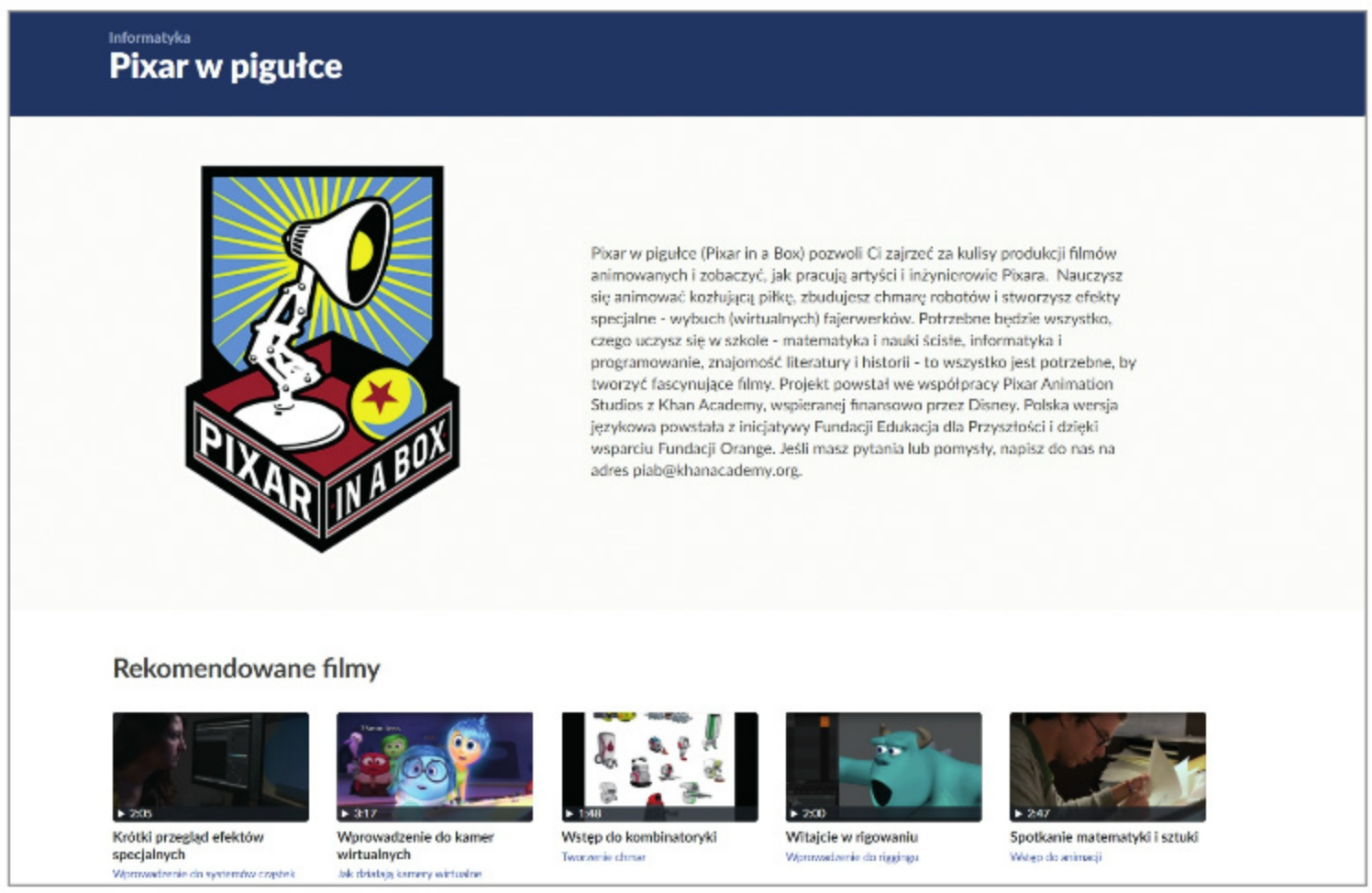
1. Na czym polegają podstawowe zadania platformy e-learningowej?
   **a.** Umożliwienie uczniom prowadzenia wybranych szkoleń.
   **b.** Nieinformowanie uczniów o ich postępach.
   **c.** Informowanie o ważnych wydarzeniach na świecie.
   **d.** Udostępnianie uczniom materiałów i aktywności.

2. Co NIE należy do podstawowych zadań platformy e-learningowej?
   **a.** Przygotowywanie kursu dla uczestników.
   **b.** Administrowanie użytkownikami.
   **c.** Udostępnianie materiałów i wspomaganie procesu kształcenia.
   **d.** Informowanie uczniów o uzyskanych ocenach oraz dostarczanie nauczycielom narzędzi do raportowania.
3. Co to jest Akademia Khana?
   **a.** To uniwersytet w Nowym Jorku.
   **b.** To organizacja edukacyjna oferująca bezpłatne kursy.
   **c.** To encyklopedia internetowa.
   **d.** To Uniwersytet Wirtualny im. Khana.

## ZADANIA DODATKOWE

### Zadanie 1. Drugi kurs

Poszukaj w Akademii Khana kursu na interesujący cię temat i spróbuj go zrealizować (zalicz co najmniej jeden moduł). Nie przeciągaj za bardzo kolejnych lekcji w czasie. Jako wynik przedstaw zestawienie swoich postępów i zdobyte odznaki.




### Zadanie 2. Platformy z kursami MOOC

Wyszukaj platformę oferującą bezpłatne kursy z przedmiotu, z którym masz trudności, lub dotyczące zagadnienia, o którym chcesz się dowiedzieć więcej. Przejrzyj ofertę i przygotuj krótką prezentację na ten temat.




# TWORZENIE STRON INTERNETOWYCH

Życie prywatne, społeczne i zawodowe w coraz większym zakresie przenosi się do sieci. Nawet podstawowa znajomość języka stron internetowych może okazać się pomocna, jeśli chcesz:

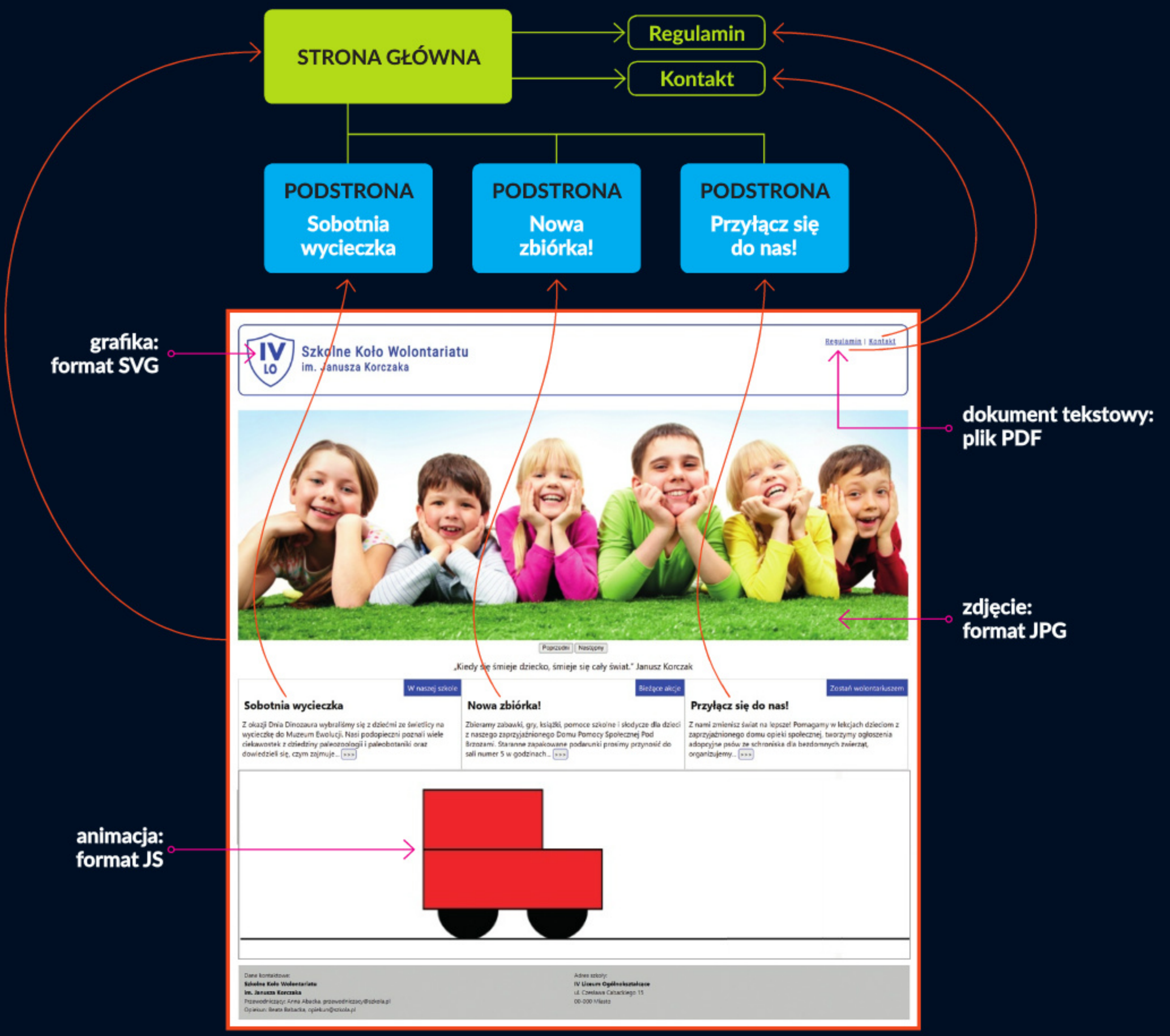
- utworzyć e-portfolio swoich prac lub e-wizytówkę,
- w pełni zarządzać stroną WWW,
- rozwijać kompetencje z dziedziny IT i zwiększyć swoje szanse na rynku pracy.

# WITRYNY I STRONY INTERNETOWE

Witryna internetowa (serwis internetowy) to zbiór stron internetowych powiązanych ze sobą za pomocą hiperłączy, udostępniony pod konkretnym adresem internetowym.

## WWW

Sieć WWW to system informacyjny, w którym dokumenty i inne zasoby internetowe są identyfikowane przez adres URL, **np. https://informatyka.wsip.pl**. Jej zasoby są przesyłane za pośrednictwem protokołu HTTP (*Hypertext Transfer Protocol*) lub połączenia szyfrowanego HTTPS (*Hypertext Transfer Protocol Secure*), przechowywane i publikowane na serwerze sieciowym oraz dostępne dla użytkowników przez przeglądarkę internetową.

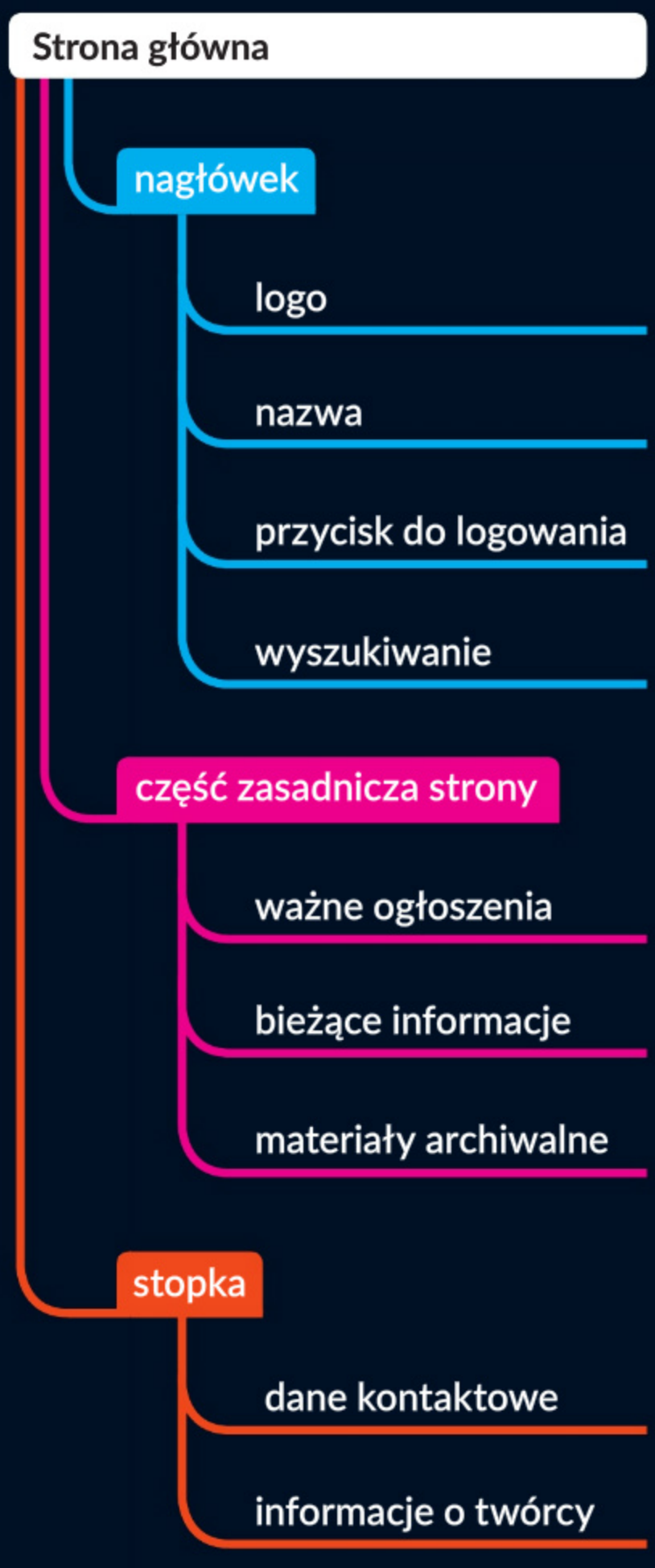



## STRONA INTERNETOWA

Strona WWW to dokument hipertekstowy udostępniany w przeglądarce. Jej podstawowym elementem jest plik tekstowy napisany w języku **HTML** (*Hypertext Markup Language*). W tym pliku osadzane są teksty, obrazy, filmy i inne pliki multimedialne. Poszczególne elementy są powiązane przez hiperłącza, które mogą prowadzić do innych stron internetowych lub poszczególnych zasobów.

## STRUKTURA WITRYNY

Każdy zbiór stron posiada swoją strukturę, którą można przedstawić na schemacie o strukturze drzewka. Pomaga to m.in. zaplanować hierarchię i wzajemne powiązanie treści (odpowiednio ułożyć informacje i zdecydować o kluczowych elementach, które powinny być lepiej widoczne i dostępne na pierwszy rzut oka) oraz nazewnictwo. Poniżej przedstawiono przykładową mapę strony.

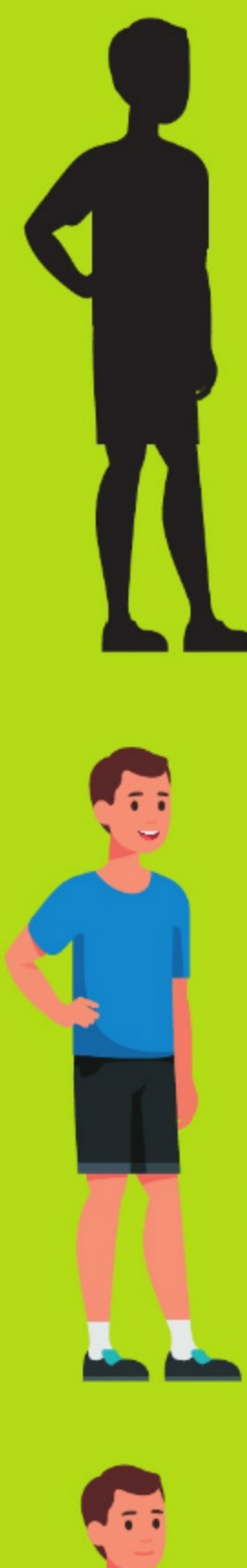



### PRACA Z INFOGRAFIKĄ

1. Jakie elementy struktury strony przedstawione na schemacie występują w serwisie twojej szkoły?
2. W jakiej kolejności należy przygotować kod na stronę? Czy można zacząć od CSS, a potem napisać kod w HTML-u? Odpowiedź uzasadnij.

## TWORZENIE STRONY WWW

Za pomocą języka HTML opisujemy strukturę informacji na stronie internetowej i nadajemy znaczenie poszczególnym fragmentom tekstu. Kolejne elementy są formatowane za pomocą kaskadowych arkuszy stylów oraz oprogramowywane za pomocą poleceń JavaScript (w celu zwiększenia interaktywności lub dynamiki danej strony). Można też użyć innego języka, takiego jak PHP lub Python.



HTML



HTML + CSS



HTML + CSS + JS




# 21. Projekt strony internetowej

- Etapy tworzenia strony WWW
- Przygotowanie projektu witryny WWW
- Projektowanie witryny na urządzenia mobilne

## ZADANIE NA START

### Jak zmienia się wygląd strony

Coraz więcej osób korzysta z przeglądarek internetowych na urządzeniach mobilnych, więc projektanci serwisów WWW dbają o to, żeby były one czytelne i funkcjonalne zarówno na małych, jak i dużych ekranach. Znajdź trzy witryny, których zawartość jest dostosowana do ekranów różnej wielkości: monitora podłączonego do komputera stacjonarnego, laptopa, tabletu i smartfona. Jakie elementy strony się zmieniają?



## PROJEKTOWANIE

Gdy tworzy się złożony projekt informatyczny, jakim jest np. witryna internetowa, warto przyjąć podejście metodyczne, tj. odpowiednio zaplanować działania i systematycznie realizować kolejne zadania zgodnie z przyjętą specyfikacją, a na koniec przeprowadzić testy.

**POSZUKAJ W INTERNECIE**

Sprawdź, czym różnią się zawody webmastera, webdevelopera i webdesignera.

Przed przystąpieniem do projektowania trzeba ustalić, co i komu chce się przekazać oraz jak zorganizować treści, żeby przekaz był czytelny.

- Jaki jest cel serwisu? Czy tworzy się go po to, aby np. popularyzować wiedzę na jakiś temat, skupić hobbystów z danej dziedziny czy też promować jakąś ideę?
- Kim jest użytkownik serwisu? Należy precyzyjnie scharakteryzować grupę docelową, wziąć pod uwagę potencjalny wiek odbiorców (np. młodzież w wieku 15–19 lat) i ich zainteresowania (np. wielbiciele jazdy konnej).
- Jakie treści ma zawierać serwis? Informacje powinny być konkretne i rzetelne, dostosowane do grupy odbiorców, np. serwis dla wielbicieli jazdy konnej powinien zawierać podstawowe informacje na ten temat i praktyczne wskazówki, natomiast na stronach biblioteki powinny się znaleźć m.in. dane teleadresowe, podstawowe informacje o bibliotece i jej strukturze, godziny dostępności, regulaminy korzystania ze zbiorów, informacje o zbiorach, opisy nowości, katalogi.
- W jaki sposób zaprezentować treści? Informacje powinny być przedstawione w sposób czytelny, ponieważ przeciętny internauta poświęca tylko kilka sekund na zorientowanie się w układzie witryny, po czym podejmuje decyzję, czy pozostać na stronie. Nawigacja powinna być intuicyjna i pozwalać na sprawny dostęp do informacji oraz na odnalezienie poszczególnych elementów.

### Ćwiczenie 1. Przygotowanie założeń witryny

Opracujcie w grupie projekt założeń strony Szkolnego Koła Wolontariatu. Przeanalizujcie powyższe pytania.

Każdy serwis powinien zawierać rzetelne, sprawdzone i w miarę możliwości aktualne informacje, a ponadto powinien być zaprojektowany zgodnie z zasadami ergonomii. Funkcjonalność z punktu widzenia użytkownika jest niezwykle ważna – wszystkie elementy należy rozmieścić w sposób przemyślany, tak aby serwis był przyjazny dla użytkownika.

**WIEM WIĘCEJ**

Można określić pięć czynników *web usability* (optymalnej użyteczności witryny):

- *learnability* (możliwość nauczenia się) – wskazuje, jak szybko użytkownik jest w stanie nauczyć się korzystania z witryny;
- *efficiency* (efektywność) – wskazuje, ile czasu potrzebuje użytkownik, aby wykonać potrzebne czynności;
- *memorability* (zapamiętywalność) – wskazuje stopień łatwości zapamiętania zasad obsługi witryny;
- *errors* (błędy) – wskazuje liczbę błędów popełnionych podczas korzystania z witryny, pozwala określić poziom poprawności jej wykonania;
- *satisfaction* (satysfakcja) – wskazuje poziom satysfakcji podczas korzystania z witryny.

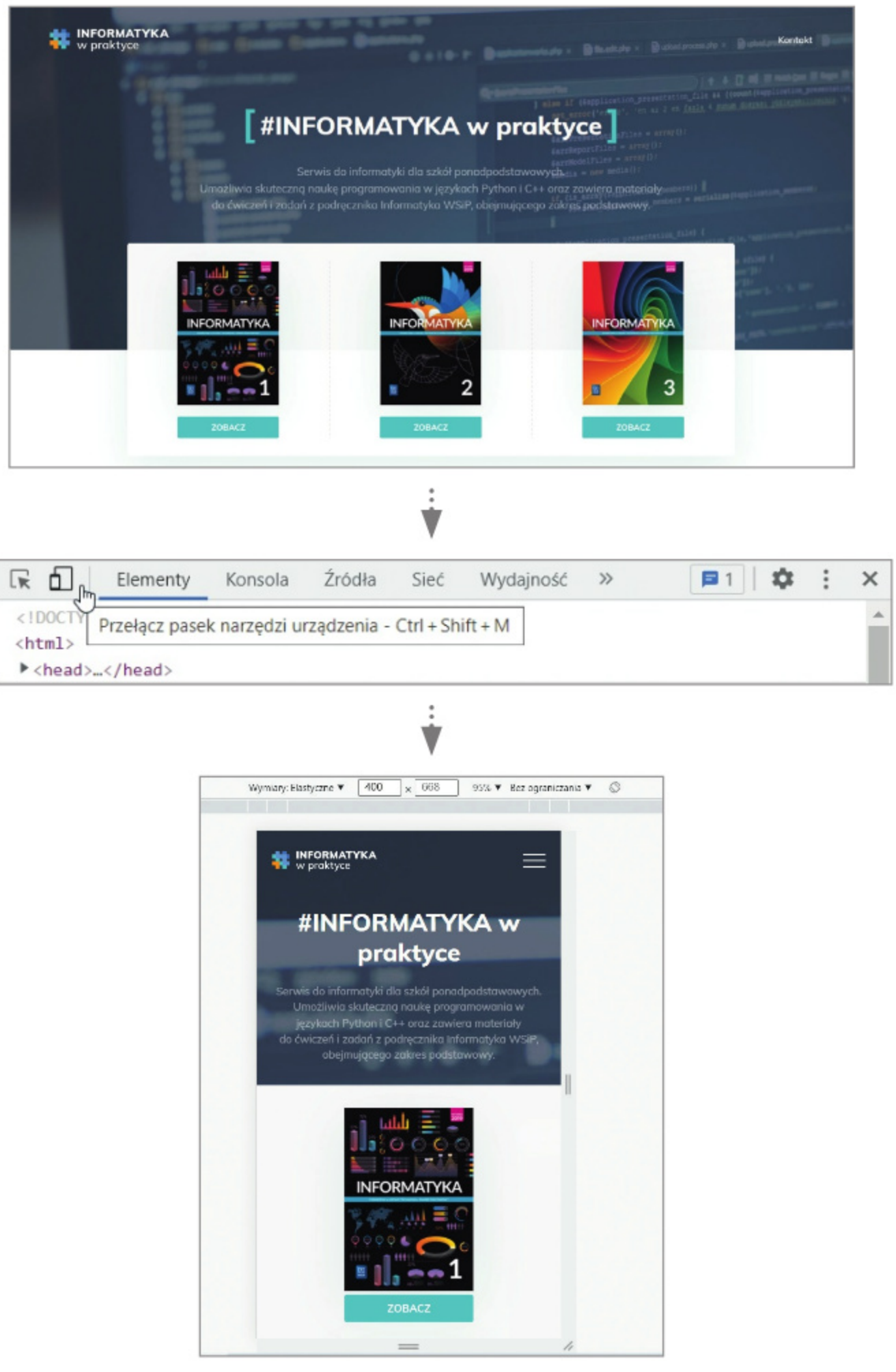
Na funkcjonalność danej witryny wpływa również możliwość dostosowania jej do wielkości ekranu urządzenia, na którym jest wyświetlana. W zależności od potrzeb można zdecydować się na projektowanie strony responsywnej lub adaptywnej.

Witrynę responsywną tworzy się jako jeden szablon, trzeba więc przemyśleć, w jaki sposób będzie się ona wyświetlała na różnych ekranach. Wygląd strony zmienia się dynamicznie i dopasowuje do rozmiaru okna przeglądarki.

W przypadku witryn adaptywnych tworzy się szablony dobrane do konkretnie określonych rozmiarów ekranów, co oznacza, że na telefonie zostanie załadowany jeden wariant strony, na tablecie – drugi, natomiast na ekranie komputera – trzeci.

**W PRAKTYCE**

Aby przekonać się, jak strona WWW będzie się prezentować na różnych urządzeniach, można ją przetestować w emulatorze przeglądarki Chrome. W tym celu należy wybrać opcję **Dostosowywanie i kontrolowanie Google Chrome** (trzy kropki z prawej strony) → **Więcej narzędzi** → **Narzędzia dla programistów**, a następnie przełączyć przycisk trybu urządzenia zaznaczony na rysunku poniżej. W menu na górnym pasku można przełączać się między różnymi typami urządzeń lub konkretnymi urządzeniami oraz dodawać ramkę urządzenia.



## Ćwiczenie 2. Przygotowanie struktury dokumentu

Opracuj projekt strony głównej Szkolnego Koła Wolontariatu (wzór poniżej). Wyodrębnij najważniejsze części serwisu i przedstaw je graficznie w formie prostokątów. Projekt przygotuj dla trzech grup urządzeń – komputerów stacjonarnych i laptopów, tabletów oraz smartfonów.




- Projekt na komputery osobiste i laptopy – ekran jest stosunkowo duży, więc elementy powinny być czytelne i zajmować w miarę możliwości znaczną część strony.



- Projekt na tablety – ekrany są trochę mniejsze w porównaniu z poprzednim projektem, więc nie wszystkie elementy zmieszczą się w poziomie.




- Projekt na smartfony – w każdym wierszu z wyjątkiem pierwszego powinien znajdować się jeden element, logo należy odpowiednio zmniejszyć.



## ŚRODOWISKO PRACY

Projektant WWW może pracować w zwykłym systemowym Notatniku, ale wygodniej jest korzystać z edytora o lepszej funkcjonalności.

Notepad++ to bezpłatny edytor tekstu oparty na licencji *GNU GPL*. Podświetla składnię, obsługuje też autouzupełnianie, powiększanie, edycję z podziałem ekranu, wyszukiwanie i (lub) zamianę ciągów znaków, a także pozwala na otwieranie kilku plików jednocześnie. Interfejs programu jest dostępny w wielu językach – również w języku polskim.

kED to polski edytor (X)HTML-a, idealny dla początkujących i średnio zaawansowanych. Jest intuicyjny w obsłudze, koloruje składnię, oferuje funkcje ułatwiające m.in. dokonywanie wstawek i podmian, wyszukiwanie i zamianę, dodawanie notatek, a poza tym pozwala na pracę z wieloma dokumentami jednocześnie (zakładki), wstawianie tabel z tekstu sformatowanego i przełączanie się między kodem a podglądem strony.

Bogatsze w funkcje są aplikacje Bluefish i Visual Studio Code. Pierwsza oferuje dodatkowo automatyczne zamykanie znaczników HTML, automatyczne generowanie galerii obrazów oraz personalizację interfejsu (np. tworzenie pasków narzędzi). Druga wspomaga pisanie kodu przez liczne podpowiedzi, a także umożliwia podgląd kodu na żywo (każda zmiana powoduje natychmiastową zmianę na stronie).

### Ćwiczenie 3. Przygotowanie środowiska pracy

Wyszukaj dalsze informacje o edytorach do tworzenia stron. Wybierz jeden z nich i go zainstaluj. Posłuży ci do przygotowania strony.

## PODSUMOWANIE

- Tworzenie stron WWW należy rozpocząć od opracowania koncepcji i projektu. Trzeba wyraźnie uświadomić sobie, do kogo strona jest skierowana i jakie informacje powinna zawierać. Dopiero potem można przejść dalej, do technicznego jej przygotowania.
- Witrynę warto przygotować tak, aby wyświetlała się czytelnie zarówno na komputerach stacjonarnych i laptopach, jak i na różnych urządzeniach mobilnych.
- Witryna responsywna dynamicznie dopasowuje swój wygląd do rozmiaru okna przeglądarki.
- Witryna adaptywna to kilka wariantów tej samej witryny, dopasowanych do konkretnych urządzeń.
- Witryny WWW można opracowywać w zwykłym systemowym Notatniku, ale wygodniej jest korzystać z edytora o większej funkcjonalności.

## PYTANIA SPRAWDZAJĄCE

1. Jakie pytania trzeba sobie zadać, gdy rozpoczyna się tworzenie projektu witryny?
2. Co trzeba wziąć pod uwagę, gdy projektuje się serwis na urządzenia stacjonarne i mobilne?
3. Czym się kierować przy wyborze narzędzi do tworzenia stron internetowych?

## PYTANIA QUIZOWE

1. Od czego należy zacząć projektowanie strony internetowej?
   **a.** Od wyboru zdjęcia na stronie głównej.
   **b.** Od określenia celu i zadań strony.
   **c.** Od korekty tekstów na stronę.
   **d.** Od wyboru programu do walidacji strony.
2. Które dokończenie zdania jest poprawne?
   Witryna responsywna
   **a.** dynamicznie dopasowuje swój wygląd do rozmiaru okna przeglądarki.
   **b.** wczytuje inny kod w zależności od urządzenia.
   **c.** jest interaktywna – zawiera elementy dynamiczne.
   **d.** inteligentnie reaguje na błędy użytkownika.
3. Których zasad należy przestrzegać podczas projektowania strony na smartfony?
   **a.** Powinien być tylko jeden wiersz, więcej się nie zmieści.
   **b.** Nie należy używać grafiki, źle się wyświetla na małym ekranie.
   **c.** Nie można używać hiperłączy, gdyż nie będą działać.
   **d.** W każdym wierszu powinien być jeden element.

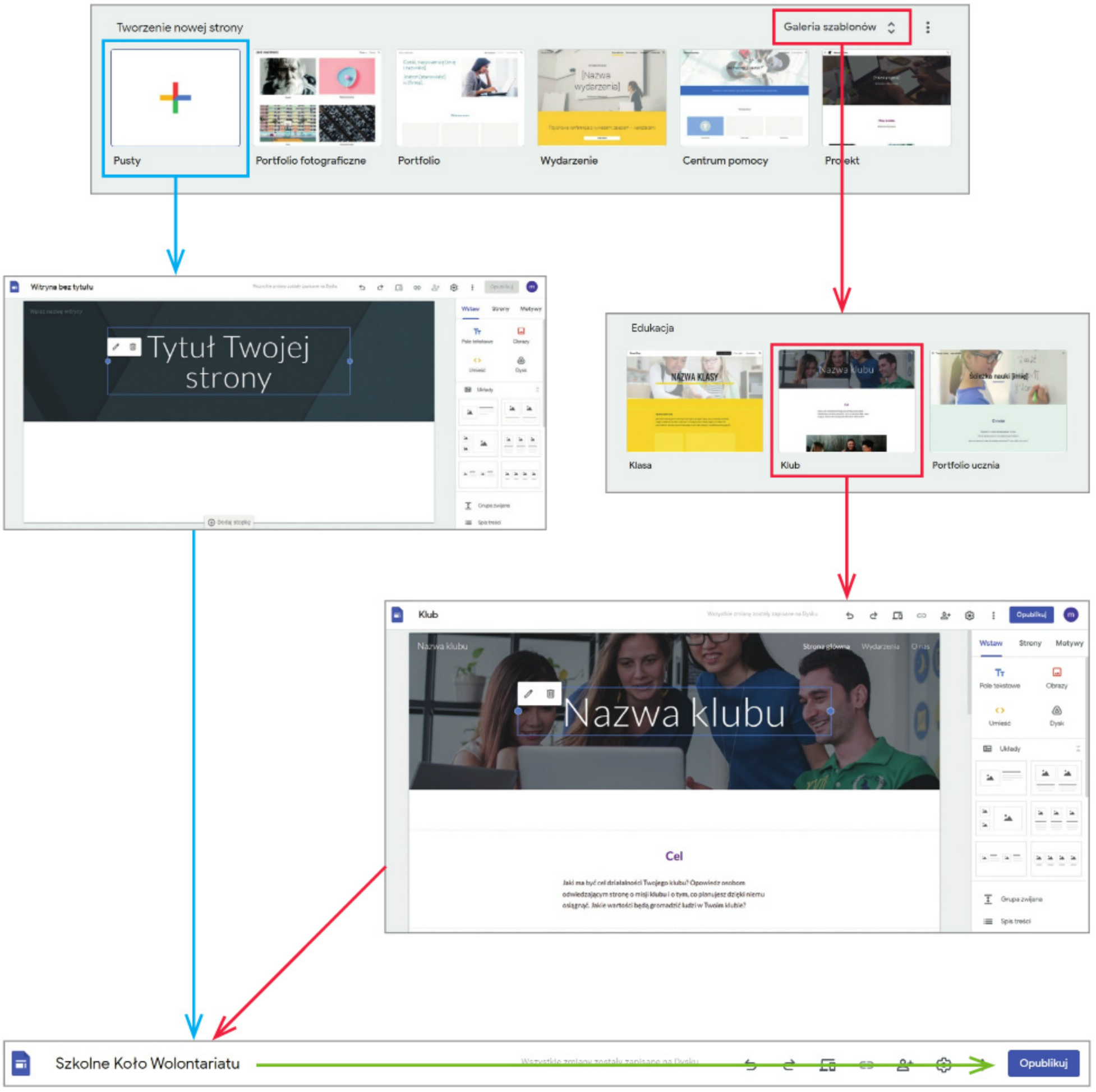
## ZADANIA DODATKOWE

### Zadanie 1. Strona w 3 minuty

Kreatory stron WWW pozwalają stworzyć nieskomplikowane witryny bez znajomości HTML-a. Gotowe szablony można przerobić na własne potrzeby – wystarczy mieć dobry pomysł na treść i oprawę graficzną oraz odpowiednio dobrać kolory. Wyszukaj serwis do szybkiego tworzenia stron internetowych. Stwórz pojedynczą stronę WWW za pomocą tego narzędzia. Zastanów się, jakie są plusy i minusy korzystania z narzędzi oferujących wykonanie skomplikowanych rzeczy w prosty sposób.

**Przykład – witryna na koncie Google**

1. Zaloguj się na swoje konto Google.
2. Wpisz adres **https://sites.google.com/new**.
3. Zacznij tworzyć stronę od zera albo wykorzystaj jeden z gotowych wzorów w galerii szablonów, a następnie skorzystaj z edytora, by dodać kolejne elementy.
4. Nadaj witrynie tytuł i ją opublikuj: kliknij przycisk **Opublikuj** (w przypadku witryny tworzonej na koncie Google nie trzeba wybierać serwera i przesyłać jej na serwer, co ułatwia aktualizacje).




## Zadanie 2. CMS

Dowiedz się, czym są i jakie funkcje oferują systemy typu CMS. Wyszukaj pięć takich systemów i krótko je scharakteryzuj.



# 22. Podstawy pracy z HTML-em

- Szablon dokumentu HTML
- Wstawianie elementów do dokumentu HTML
- Definiowanie głównych składowych strony WWW

## ZADANIE NA START

**Który język do czego służy**

Dopasuj definicję języka do jego skrótowej nazwy.

**I.** HTML **II.** CSS **III.** JS

A. Język, w którym opisuje się formę prezentacji (formatowanie) poszczególnych elementów dokumentu.
B. Język, za pomocą którego opisuje się strukturę informacji na stronie internetowej i nadaje znaczenie poszczególnym fragmentom tekstu; przy jego użyciu można m.in. tworzyć akapity tekstu, hiperłącza, nagłówki, listy oraz osadzać multimedia.
C. Język, dzięki któremu można zdefiniować na stronie elementy dynamiczne, służące do zapewnienia interakcji z użytkownikiem przez reagowanie na zdarzenia (np. kliknięcie), tworzenie złożonych efektów wizualnych (np. animacji) oraz obsługę danych wprowadzanych w formularzach (np. sprawdzanie poprawności danych).




## OGÓLNA STRUKTURA DOKUMENTU

Po przygotowaniu projektu strony można się zająć definiowaniem jej struktury. Należy zacząć od szablonu strony, a potem dodawać kolejne elementy – nagłówek i stopkę, rysunki, odnośniki oraz karty, na których będą wyświetlane informacje. Na tym etapie trzeba się koncentrować na poprawnym wprowadzeniu danych i zbudowaniu struktury dokumentu, a nie na jego wyglądzie.

```
1.  <!DOCTYPE html>
2.  <html lang="<!--język-->">
3.  <head>
4.    <meta charset="<!--kodowanie znaków-->">
5.    <meta name="viewport" content="<!--sposób wyświetlania strony-->">
6.    <title><!--tytuł strony--></title>
7.  </head>
8.
9.  <body>
10.   <header>
11.   <!--nagłówek-->
12.   </header>
13.   <!--część zasadnicza strony-->
14.   <footer>
15.   <!--stopka-->
16.   </footer>
17. </body>
18. </html>
```

W HTML-u komentarz umieszcza się między znakami <!-- i -->.

**Rys. 1.** Struktura dokumentu HTML

Deklarację języka na danej stronie formułuje się za pomocą atrybutu **lang** znacznika **<html>**. W przypadku języka polskiego wystarczy zadeklarować **pl**.

**WIEM WIĘCEJ**

Ponieważ niektóre języki są używane w kilku krajach (np. francuski czy niemiecki), niekiedy konieczne jest dodatkowo zadeklarowanie kraju, w którym używa się danego języka, np. **fr-FR** (francuski, Francja), **fr-CA** (francuski, Kanada), **de-CH** (niemiecki, Szwajcaria), **de-AT** (niemiecki, Austria).

Dodatkowe informacje dla wyszukiwarek wprowadza się za pomocą metatagów w sekcji **head**. Zwykle stosuje się następujące metaznaczniki:

- **<meta charset=" ">** – deklaruje kodowanie znaków (każdy plik jest zapisany w pewnym standardzie, który w określony sposób zapisuje znaki narodowe, np. *ą*, *ę*, *ń*, oraz inne znaki specjalne); obecnie przeważa standard UTF-8 obsługujący wiele języków jednocześnie;
- **<meta name="description" content=" ">** – opisuje zawartość serwisu lub strony (opis powinien mieć ok. 160 znaków);
- **<meta name="viewport" content=" ">** – deklaruje sposób wyświetlania strony na urządzeniach mobilnych, m.in. szerokość i wysokość wyświetlanego obszaru strony (np. **"width=320, height=480"** albo **"width=device-width"**), domyślne przybliżenie strony (np. **"initial-scale=1"**), minimalne i maksymalne przybliżenie (np. **"minimum-scale=1, maximum-scale=10"**);

- **<meta name="keywords" content=" ">** – zawiera słowa kluczowe określające tematykę strony (obecnie ten znacznik stracił na znaczeniu – wyszukiwarki Google, Yahoo czy Bing samodzielnie ustalają listę słów najlepiej odpowiadających treści strony).

Informację dla wyszukiwarki na temat tytułu strony, który powinien być widoczny na pasku przeglądarki internetowej, umieszcza się za pomocą znacznika **<title>**. Tytuł powinien być zwięzły i odnosić się do zawartości strony/serwisu.

```
1. <!DOCTYPE html>
2. <html lang="pl">
3. <head>
4. <meta charset="utf-8">
5. <meta name="description" content="Strona poświęcona powieści H.G.
Wellsa Wehikuł czasu">
6. <meta name="viewport" content="width=device-width, initial-scale=1">
7. <meta name="keywords" content="Wells, H.G. Wells, wehikuł, wehikuł
czasu, sf, science fiction">
8. <title>Wszystko o Wehikule czasu</title>
9. </head>
```

**Rys. 2.** Przykładowa sekcja **head** strony poświęconej powieści *Wehikuł czasu*

Niektóre zagadnienia omawiane podczas tej i następnej lekcji w podręczniku zostały tylko zasygnalizowane, ponieważ szczegółowo omawia się je na kursie **Wstęp do HTML/CSS: Tworzenie stron internetowych**, który powinien być zrealizowany w ramach lekcji 20. W razie potrzeby wróć do kursu albo poszukaj odpowiednich informacji w sieci.

## Ćwiczenie 1. Szablon

Przygotuj szablon strony głównej Szkolnego Koła Wolontariatu (SKW) – utwórz podstawową strukturę strony w języku polskim. Strona powinna być przeznaczona na urządzenia mobilne (mieć 100 % szerokości obszaru ekranu oraz pozostać w skali 1 : 1 z możliwością skalowania).

1. W wybranym edytorze odwzoruj szablon struktury dokumentu HTML z rys. 1.
2. Uzupełnij odpowiednio szablon, tj. zadeklaruj język, kodowanie znaków oraz sposób wyświetlania strony według wskazówek w poleceniu, a następnie podaj tytuł strony.
3. Zapisz gotowy dokument jako **index.html**.

■ Szablon strony głównej koła może wyglądać tak, jak przedstawiono na stronie obok.

```
1. <!DOCTYPE html>
2. <html lang="pl">
3. <head>
4.   <meta charset="utf-8">
5.   <meta name="viewport" content="width=device-width, initial-scale=1">
6.   <title>Szkolne Koło Wolontariatu</title>
7. </head>
8.
9. <body>
10.   <header>
11.   </header>
12.   <footer>
13.   </footer>
14. </body>
15. </html>
```

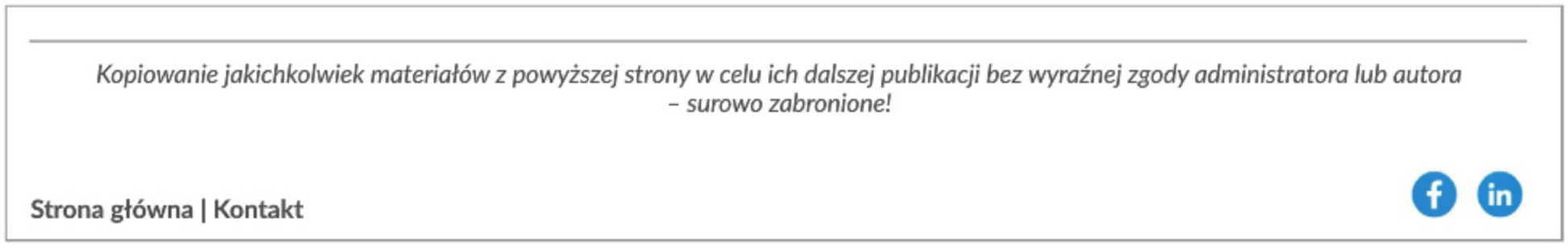
## NAGŁÓWEK I STOPKA

W nagłówku wydzielonym znacznikami **<header></header>** zwykle umieszcza się nazwę instytucji, logo, krótką informację, wyszukiwarkę, a także elementy nawigacyjne, np. odnośnik do strony głównej albo formularza kontaktowego.




**Rys. 3.** Przykładowy nagłówek strony poświęconej powieści *Wehikuł czasu*

Znaczniki **<footer></footer>** wydzielają obszar przeznaczony na stopkę dokumentu lub sekcji (w jednej sekcji może się znajdować tylko jedna stopka). Stopka powinna być dopasowana do layoutu i charakteru strony (serwisu): inaczej powinna wyglądać na stronie firmowej, a inaczej na stronie prywatnej czy też blogu. Najczęściej umieszcza się tam datę powstania witryny, nazwę i dane firmy lub organizacji, imię i nazwisko właściciela witryny, nazwę twórcy strony i informacje o prawach autorskich albo odnośniki do powiązanych dokumentów (np. warunków użytkowania serwisu), stron w ramach serwisu (rodzaj menu) lub innych serwisów (np. społecznościowych).




**Rys. 4.** Przykładowa stopka strony poświęconej powieści *Wehikuł czasu*

Odnośniki wewnętrzne mogą się odwoływać do innego miejsca w dokumencie lub otwierać dokument. Odnośniki zewnętrzne przenoszą użytkownika na inne strony. Domyślnie każdy link ustawiony jest tak, aby otwierał się w tej samej karcie. Aby otworzyć plik w osobnym oknie, należy dodatkowo zastosować atrybut **target="_blank"**.

```
1. <a href="#recenzje">Recenzje</a>
2. <a href="moj_przeklad.pdf"
   target="_blank">Mój przekład</a>
3. <a href="https://wolnelektury.pl/
   katalog/lektura/wells-wehikul-
   czasu/" target="_blank">Przekład
   w Wolnych Lekturach</a>
```

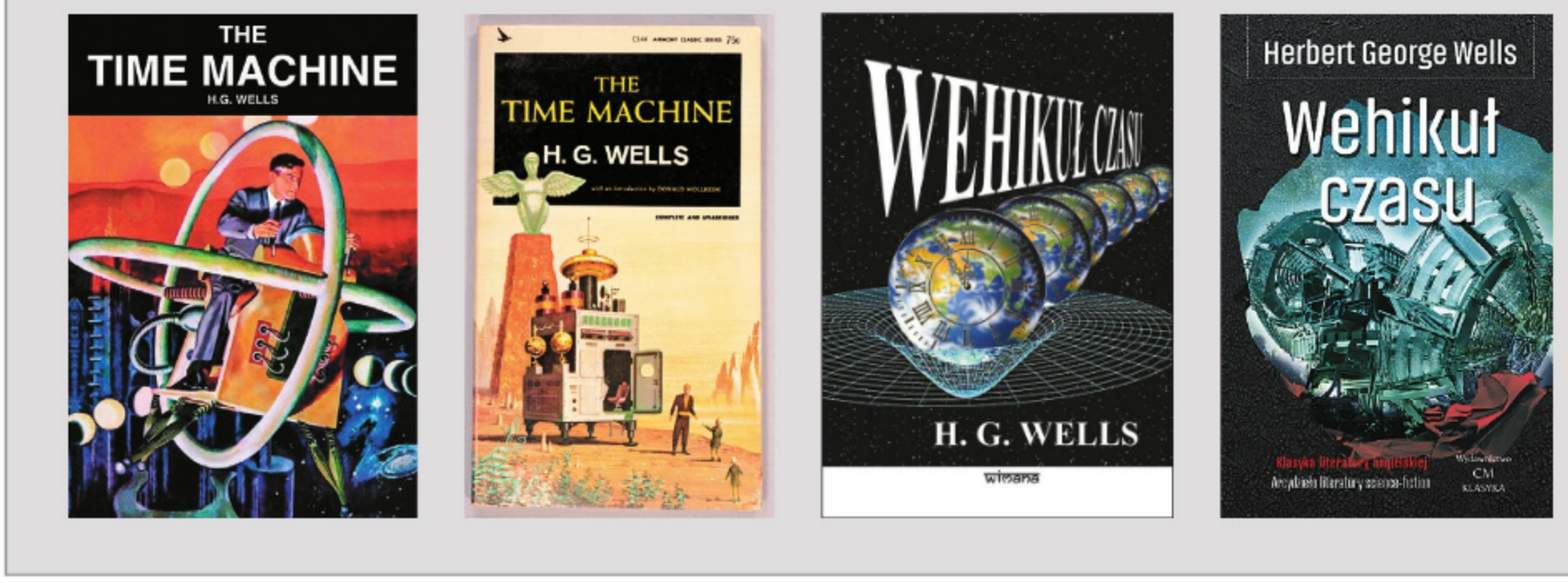
Link wewnętrzny pomagający w nawigacji po stronie. Wymaga oznaczenia danego elementu unikatową nazwą (**id**), np. recenzje. W odnośniku ustalone **id** poprzedza się krzyżykiem #.

Link wewnętrzny otwierający dokument w osobnej karcie.

Link zewnętrzny otwierający stronę w osobnej karcie.

**Rys. 5.** Przykładowe linkowanie: linki wewnętrzne (wiersze 1 i 2) i link zewnętrzny (wiersz 3)

Aby wstawić obrazek, np. logo, należy użyć znacznika **<img>**. Oprócz źródła **src** trzeba wstawić tekst alternatywny **alt** uaktywniany wtedy, gdy z jakichś powodów obrazek nie może zostać wyświetlony lub gdy strona analizowana jest przez automatyczny czytnik, np. wspomagający pracę niewidomych.




```
<img src="grafika/wehikul.png" alt="okładki powieści Wehikuł czasu">
```

**Rys. 6.** Cztery okładki wstawione na stronę jako jeden obrazek i odpowiadający mu kod HTML

Do pogrupowania i ustrukturyzowania elementów w nagłówku, stopce lub innej części strony (serwisu) można użyć znaczników ** **. Często stosuje się je wraz z atrybutami **class** (klasa obiektu) lub **id** (niepowtarzalna nazwa). Znacznik ** ** jest elementem blokowym. Służy np. do wyodrębnienia kolumny tekstu, pojedynczej karty czy menu.

**W PRAKTYCE**

Element mający unikalny identyfikator (**`id`**) służy do linkowania do konkretnej części dokumentu lub do opisywania stylu tego elementu w CSS. Jeśli elementów do ostylowania jest więcej, zamiast atrybutu **`id`** stosuje się klasę (atrybut **`class`**).

Treści zawarte w nagłówku i stopce można wyróżnić i sformatować za pomocą znaczników tworzących **nagłówki, zwane też tytułami lub śródtytułami** (termin *nagłówek* może oznaczać m.in. podział treści dokumentu oraz nagłówek w szkielecie dokumentu, tj. **head**; są to osobne zagadnienia i nie należy ich mylić). W języku HTML jest do dyspozycji sześć poziomów nagłówków, gdzie **`<h1></h1>`** oznacza nagłówek pierwszego stopnia, a **`<h6></h6>`** – nagłówek szóstego stopnia, ostatni w hierarchii.

**Aby wizualnie oddzielić fragment tekstu** od reszty treści, np. poprzez pogrubienie, stosuje się znaczniki **`<strong></strong>`**.

**Aby przenieść tekst do kolejnego wiersza**, np. podczas pisania adresu, należy na końcu wiersza, który ma zostać przełamany, wstawić pojedynczy znacznik **`<br>`**.

## Ćwiczenie 2. Nagłówek i stopka

Wypełnij nagłówek i stopkę strony głównej Szkolnego Koła Wolontariatu odpowiednią treścią (wzór poniżej). W nagłówku umieść nazwę koła, logo szkoły oraz linki – do regulaminu (otwierany w osobnym oknie) i do danych kontaktowych na dole strony. W stopce umieść dane kontaktowe.




Regulamin | Kontakt

**Szkolne Koło Wolontariatu**

**im. Janusza Korczaka**

Dane kontaktowe:
**Szkolne Koło Wolontariatu**
**im. Janusza Korczaka**
Przewodniczący: Anna Abacka, przewodniczacy@szkola.pl
Opiekun: Beata Babacka, opiekun@szkola.pl
Adres szkoły:
**IV Liceum Ogólnokształcące**
ul. Czesława Cabackiego 15
00-000 Miasto

1. Uzupełnij sekcję **header** w dokumencie HTML z ćwiczenia 1.

- Dodaj kod odpowiadający za wyświetlanie w przeglądarce logo lub tekstu alternatywnego. Zastosuj atrybut **`id`** – w miarę rozbudowywania projektu ułatwi to odnoszenie się do tego obiektu.
- Wyodrębnij blok tekstu zawierający odnośniki do regulaminu i danych kontaktowych. Zastosuj atrybut **`class`** – na dalszym etapie realizacji projektu ułatwi to ostylowanie tej części strony.
- Dodaj kody dotyczące linkowania wewnątrz dokumentu i otwierania PDF-a w osobnym oknie. Rozdziel odnośniki znakiem I.
- Jeśli nazwa koła jest dłuższa, np. zawiera imię i nazwisko patrona, zapisz ją w dwóch wierszach i dodaj nagłówki, które zróżnicują wielkość tekstu.

- Ponieważ poszczególne elementy nagłówka i stopki będą ułożone obok siebie, wstaw blok, który zamknie wszystko i spowoduje, że następne dane znajdą się poniżej.

```
<header>
    <img id="logo" src="grafika/logo.png" alt="logo_szkoły">
    <div class="info">
        <a href="dokumenty/regulamin.pdf" target="_blank">Regulamin</a>
        |
        <a href="#kontakt">Kontakt</a>
    </div>
    <h1>Szkolne Koło Wolontariatu</h1>
    <h2>im. Janusza Korczaka</h2>
    <div class="koniec"></div>
</header>
```

2. Uzupełnij sekcję **footer** w dokumencie HTML z ćwiczenia 1.

- Wyodrębnij dwa bloki tekstu zawierające – odpowiednio – dane kontaktowe koła oraz adres szkoły. W przypadku bloku stanowiącego punkt nawigacyjny zastosuj atrybut **id**.
- Wyróżnij nazwę koła i nazwę liceum.
- Przełam tekst – odpowiednio – w danych kontaktowych i w adresie.
- Dodaj blok zamykający.

```
<footer>
    <div id="kontakt">
      Dane kontaktowe:<br>
      <strong>Szkolne Koło Wolontariatu<br>
      im. Janusza Korczaka</strong><br>
      Przewodniczący: Anna Abacka, przewodniczacy@szkola.pl<br>
      Opiekun: Beata Babacka, opiekun@szkola.pl<br>
    </div>
    <div>
      Adres szkoły:<br>
      <strong>IV Liceum Ogólnokształcące</strong><br>
      ul. Czesława Cabackiego 15<br>
      00-000 Miasto
    </div>
    <div class="koniec"></div>
</footer>
```

3. Zapisz plik i wyświetl stronę w przeglądarce.

## ZASADNICZA CZĘŚĆ STRONY

Do grupowania i strukturyzowania elementów strony można oprócz znaczników ** ** wykorzystać znaczniki ****. Różnica między **** a ** ** polega na tym, że **** jest elementem liniowym, za pomocą którego można np. wyróżnić w paragrafie fragment tekstu innym kolorem lub utworzyć przycisk. Paragrafy tworzy się za pomocą znacznika **<p>** oraz odpowiadającego mu znacznika zamykającego. Dodatkowo można dodać znaczniki typowe dla HTML5:

- **<nav></nav>** – wyznaczają obszar przeznaczony na elementy nawigacyjne (menu);
- **<section></section>** – oddzielają odrębne bloki dokumentu (sekcje);
- **<article></article>** – wyznaczają oddzielną, niezależną część dokumentu;
- **<aside></aside>** – wyznaczają obszar na elementy dodatkowe, np. reklamy.

**POSZUKAJ W INTERNECIE**

Na stronie **https://www.w3schools.com** dostępny jest spis poleceń języka HTML wraz z przykładami ich użycia. Poszukaj podobnych stron w języku polskim.

### Ćwiczenie 3. Wstawianie elementów na stronę

Uzupełnij treść strony głównej SKW – dodaj obrazek, cytat oraz karty zawierające wydarzenia przeszłe i planowane oraz informację o tym, jak dołączyć do koła.

1. Wyodrębnij blok zawierający obraz.

- Zastosuj atrybut **class**, który będzie oznaczał klasę elementów umieszczonych w jednym wierszu.

```
1. <div class="row">
2.     <img src="grafika/dzieci.png" alt="Uśmiechnięte dzieci">
3. </div>
```

- Dodaj kod odpowiadający za wyświetlanie w przeglądarce obrazka lub tekstu alternatywnego.

2. Wyodrębnij blok zawierający cytat.

- Zastosuj atrybut **class**, który będzie oznaczał klasę elementów umieszczonych w jednym wierszu, oraz atrybut **id**, który później pozwoli łatwo ostylować ten element strony.

```
5. <div class="row" id="cytat">
6.     "Kiedy się śmieje dziecko, śmieje się cały świat." Janusz Korczak
7. </div>
```

3. Wyodrębnij trzy bloki i nadaj im klasę, za pomocą której będzie można później odpowiednio sformatować wszystkie karty w serwisie.

```
8. <div class="row">
9.      <div class="karta"></div>
10.     <div class="karta"></div>
11.     <div class="karta"></div>
12.     <div class="koniec"></div>
```

4. W każdym bloku zawierającym kartę wyodrębnij blok z klasą oznaczającą winietkę w prawym górnym rogu karty, wpisz tytuł karty opisany odpowiednim nagłówkiem i fragment tekstu oraz wstaw i oznacz symbol, który później stanie się przyciskiem pozwalającym przejść na stronę artykułu.

```
9.      <div class="karta">
10.         <div class="info">W naszej szkole</div>
11.         <h2>Sobotnia wycieczka</h2>
12.         <p>Z okazji Dnia Dinozaura wybraliśmy się z dziećmi
            ze świetlicy na wycieczkę do Muzeum Ewolucji. Nasi
            podopieczni poznali wiele ciekawostek z dziedziny
            paleozoologii i paleobotaniki oraz dowiedzieli się, czym
            zajmuje... <span class="przycisk"> >>>
            </span></p>
13.     </div>
```

5. Zapisz plik i wyświetl stronę w przeglądarce.

## Ćwiczenie 4. Kolejne strony

Dodaj do serwisu SKW dwie strony zaprojektowane na wzór strony głównej. Powinny mieć taki sam nagłówek, różny od nagłówka strony głównej, i stopkę identyczną z tą na stronie głównej. Wykorzystaj tabele oraz listy.

**Tabele**

```
<table>
  <tr><th>nagłówek 1</th>
          <th>nagłówek 2</th></tr>
  <tr><td>dana1</td>
         <td>dana2</td>
</tr>
</table>
```

**Podstawowe wyróżnienia**

```
<strong> – pogrubienie
<em> – pochylenie
<u> – podkreślenie
<sup>, <sub> – indeks górny i dolny
<br> – przejście do nowego wiersza
<hr> – linia pozioma
```

**Lista wypunktowana**

```
<ul>
  <li>pierwsza opcja</li>
  <li>druga opcja</li>
  <li>trzecia opcja</li>
</ul>
```

**Lista numerowana**

```
<ol>
  <li>pierwsza opcja</li>
  <li>druga opcja</li>
  <li>trzecia opcja</li>
</ol>
```

## PODSUMOWANIE

- Odpowiednia organizacja strony wpływa na czytelność przekazu. Przemyślana nawigacja zapewnia szybki dostęp do zamieszczonych informacji.
- Warto stosować przyjęte konwencje. Nagłówek strony powinien zawierać logo i nazwę instytucji, a także elementy nawigacyjne. W stopce należy umieścić dane kontaktowe i regulacje prawne.
- Wszystkie elementy powinny być rozmieszczone w sposób intuicyjny, tak aby serwis był jak najbardziej funkcjonalny. Ułatwiają to znaczniki, np. **`<div>`** służy do wydzielania bloków.

## PYTANIA SPRAWDZAJĄCE

1. Jakie elementy powinny znaleźć się w nagłówku strony, a jakie w stopce?
2. Czym różnią się znaczniki **`<div>`** i **`<span>`**?
3. Jak stworzyć link do dokumentu zewnętrznego, a jak do innego miejsca na stronie?

## PYTANIA QUIZOWE

1. Jaki napis z hiperłączem będzie widoczny po umieszczeniu na stronie kodu **`<a href="dokument.pdf" target="_blank">dokument</a>`**?
   **a.** blank
   **b.** _blank
   **c.** dokument.pdf
   **d.** dokument
2. Do czego służy znacznik **`<span>`**?
   **a.** Do wyróżniania fragmentu tekstu.
   **b.** Do wyznaczania akapitów.
   **c.** Do wstawiania hiperłączy.
   **d.** Do dołączania grafiki.
3. Które dokończenie zdania jest poprawne?
   Znacznik **`<img>`** służy do osadzania obrazków na stronie i
   **a.** można go użyć w dokumencie tylko raz.
   **b.** jest dozwolony tylko w nagłówku.
   **c.** jest dozwolony tylko w stopce.
   **d.** nie wymaga zamknięcia.

## ZADANIE DODATKOWE

### Zadanie 1. Multimedia

Dodaj do swojego serwisu elementy multimedialne – nagranie (np. wywiad), krótki filmik lub animację. Pamiętaj o prawach autorskich.

# POZYCJONOWANIE I OPTYMALIZACJA STRON WWW

**Po wpisaniu w wyszukiwarce hasła internauta otrzymuje listę stron. Ale kolejność ich wyświetlania nie jest przypadkowa – ustala ją algorytm zastosowany w wyszukiwarce. Ranking wyszukiwania wyników zmienia się dynamicznie.**

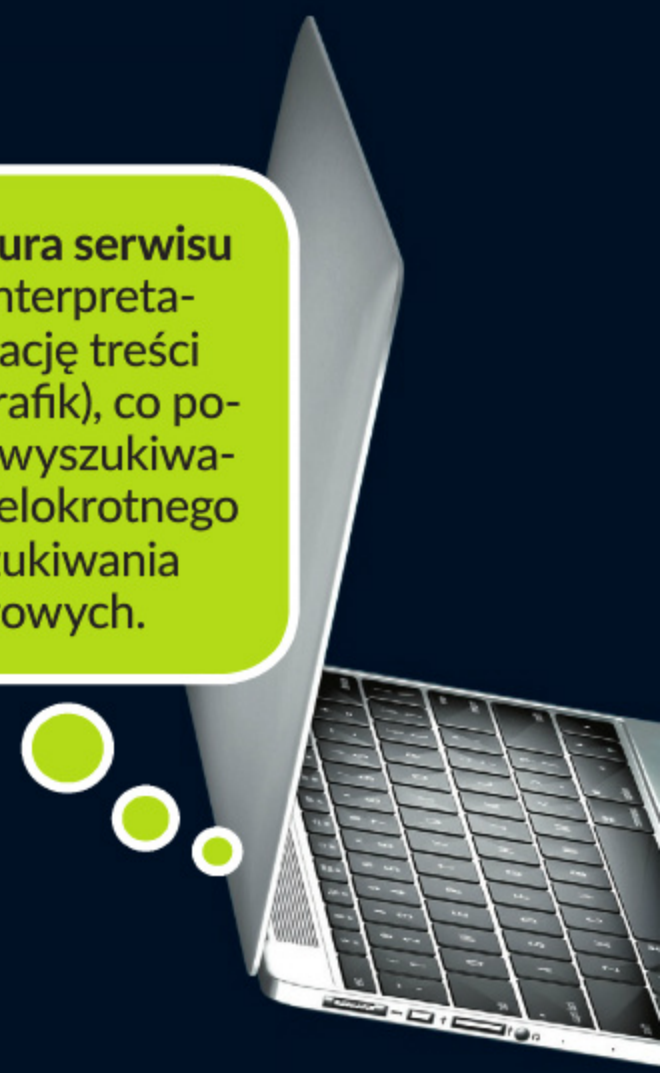
**Pozycjonowanie strony** to działania mające na celu zwiększenie jej widoczności w wynikach wyszukiwania:

- tworzenie wartościowych treści,
- pozyskiwanie użytkowników zainteresowanych tematyką strony,
- motywowanie użytkowników do udostępniania i cytowania treści strony

oraz

- **optymalizacja** (*Search Engine Optimization*, SEO), czyli dostosowanie strony do odpowiedniego analizowania przez wyszukiwarki.

Uporządkowana **struktura serwisu** umożliwia właściwą interpretację i poprawną indeksację treści (np. wyrazów, filmów, grafik), co pozwala na błyskawiczne wyszukiwanie bez konieczności wielokrotnego analizowania i przeszukiwania dokumentów źródłowych.



## OPTYMALIZACJA

### TREŚCI

- oryginalna treść
- słowa kluczowe
- hierarchia nagłówków
- czytelna struktura (menu, treść)
- wewnętrzne linki

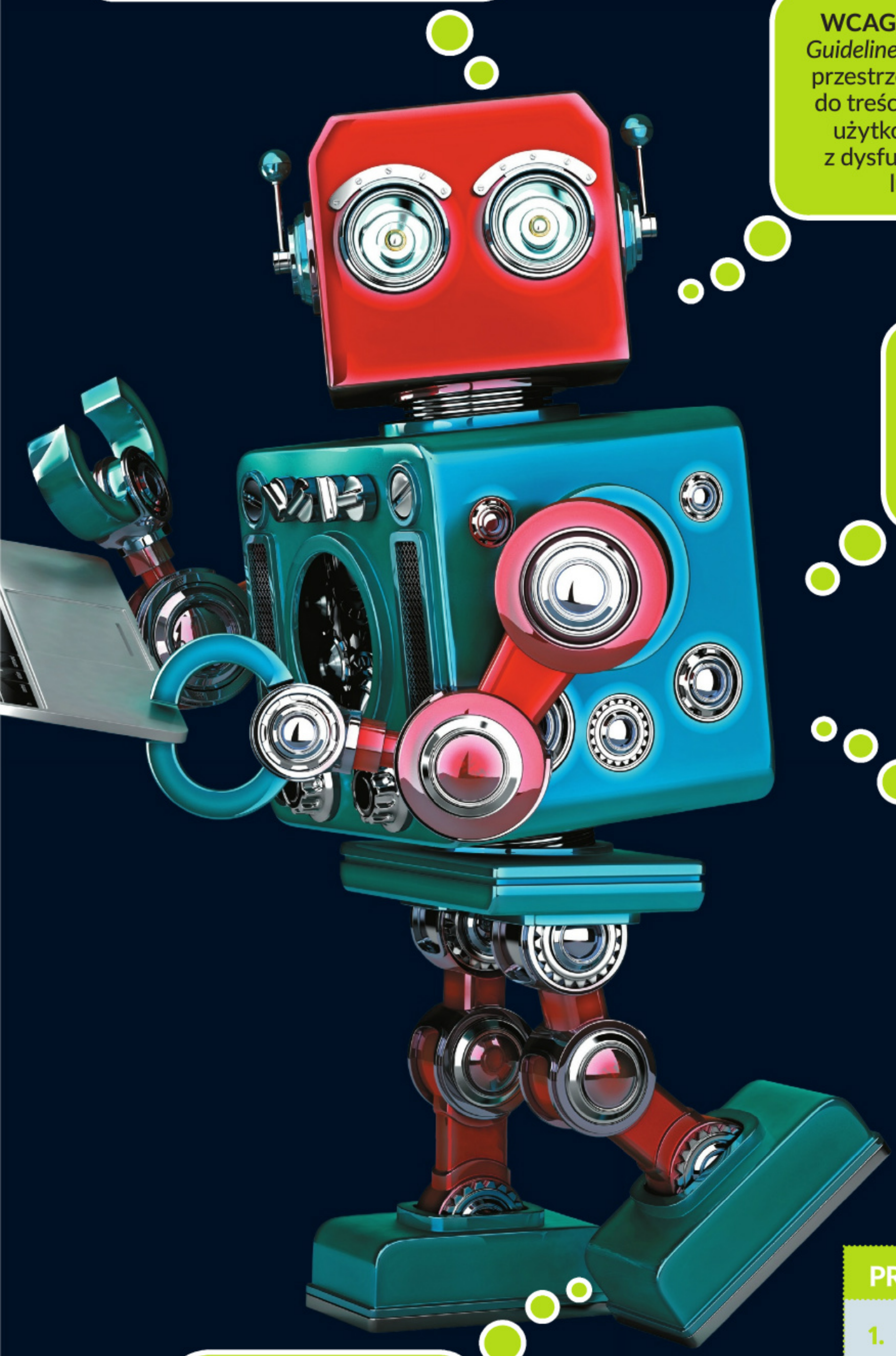
### KODU

- zgodność ze standardem W3C wyświetlania na różnych urządzeniach
- szybkość ładowania witryny (zmniejszanie objętości kodu, liczby wczytywanych plików, optymalizacja grafiki)
- automatyczne analizowanie witryny (hierarchia nagłówków, metatagi, przyjazne adresy URL)

### PROFILU LINKÓW PRZYCHODZĄCYCH

- liczba linków oraz domen, z których pochodzą linki
- popularność stron, na których są zamieszczone linki do danej strony

**Tytuł strony** to jeden z ważniejszych czynników technicznych, na podstawie którego wyszukiwarka ocenia tematykę strony. Jest wyświetlany w wynikach wyszukiwania.



**WCAG** (*Web Content Accessibility Guidelines*) to zasady, których należy przestrzegać, aby zapewnić dostęp do treści możliwie szerokiej grupie użytkowników, w tym osobom z dysfunkcjami, np. niewidomym lub słabowidzącym.

**Opisy meta** ułatwiają automatyczną analizę strony i wpływają na treść opisu wyświetlanego w wynikach wyszukiwania.

**Opisy alternatywne do zdjęć** nie tylko są pomocą dla osób słabowidzących lub niewidzących, lecz także wspomagają proces automatycznego analizowania grafiki.

Odpowiednio **skompresowane grafiki** skracają czas ładowania strony.

## PRACA Z INFOGRAFIKĄ

1. Co możemy zrobić, by nasza strona była wyżej w rankingu wyszukiwania?
2. Po co są zasady WCAG?



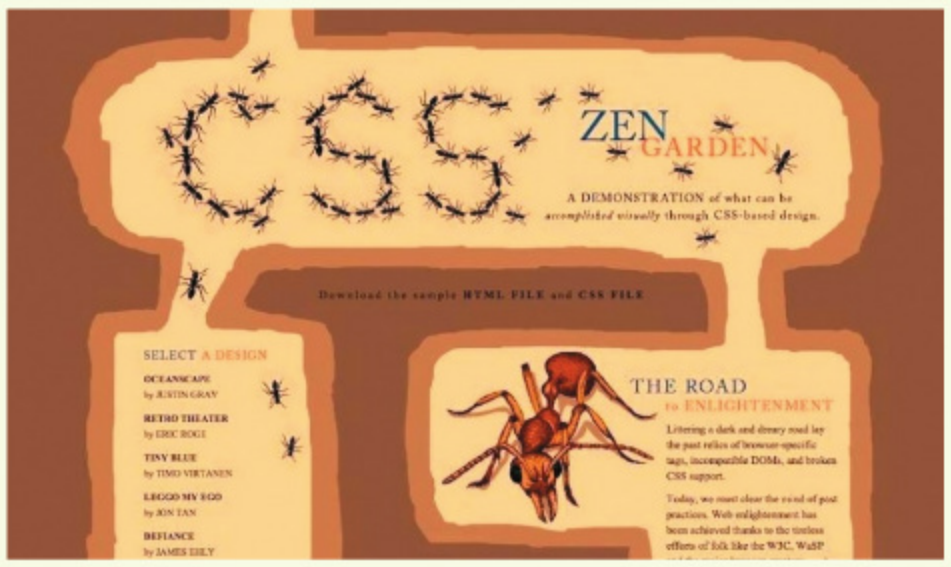

# 23. Podstawy pracy z CSS-em

- Struktura i treść a wygląd dokumentu
- Projektowanie wyglądu typowych elementów strony
- Dostosowywanie wyglądu strony do różnych urządzeń

## ZADANIE NA START

### Ogród CSS

Na stronie **www.csszengarden.com** pokazano, jak różnie można zaprojektować stronę za pomocą arkusza stylów. Każdy projektant wykorzystuje ten sam kod HTML, różnice wizualne między projektami wynikają z arkusza stylów oraz grafik, do których odnosi się arkusz. Przyjrzyj się projektom i wskaż ich charakterystyczne cechy.













## ABC KASKADOWYCH ARKUSZY STYLÓW

Informacje dotyczące wyglądu dokumentu HTML można umieszczać wewnątrz dokumentu HTML lub w osobnym pliku. Reguły wpisane dołącza się do poszczególnych znaczników HTML. Ponieważ określają one sposób wyświetlania wyłącznie tego elementu, którego dotyczy dany znacznik, używanie stylu wpisanego dla większej liczby znaczników jest kłopotliwe i skutkuje powstawaniem obszernego kodu. Reguły osadzone umieszcza się w części nagłówkowej dokumentu HTML – w ten sposób definiuje się styl

wszystkich elementów danego typu. Jeśli witryna składa się z kilku stron, trzeba powtórzyć definicję stylu dla każdej z nich. Reguły zewnętrzne umieszcza się w osobnym pliku tekstowym z rozszerzeniem CSS – do tak zdefiniowanych stylów można się odwoływać na wszystkich stronach tworzących daną witrynę. W części nagłówka dokumentu HTML należy wpisać adres pliku stylu z wykorzystaniem polecenia:
`<link rel="stylesheet" type="text/css" href="nazwa_pliku.css">`.

Każda reguła określająca styl danego elementu dokumentu składa się z selektora oraz z deklaracji otoczonej nawiasami klamrowymi. Selektory można grupować, a także określać ich wzajemne zależności. Selektorem może być znacznik języka HTML (np. nagłówek **h1**), atrybut identyfikatora elementu (**id**), czyli jego niepowtarzalna nazwa (np. **#cytat**), albo klasa elementu (np. **.karta**). Deklaracje wyznaczają formatowanie i składają się z nazwy właściwości oraz jej wartości.

```
1. p {text-align: justify;}
2. #wstep {font-size: 1.2em;}
3. .uwaga {background-color: #ff0000;}
```

wyjustowanie tekstu we wszystkich akapitach

powiększenie czcionki w elemencie o id #wstep

ustawienie czerwonego koloru tła w elementach klasy uwaga

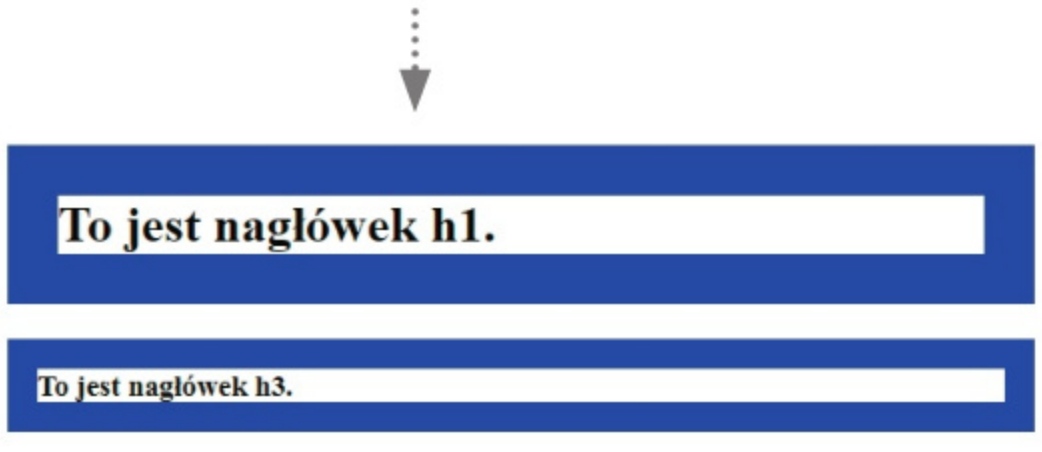
**Rys. 1.** Przykładowe reguły

**W PRAKTYCE**

Podczas określania wartości liczbowych można się posługiwać pikselami, procentami albo jednostkami relatywnymi **em** i **rem**. Jednostka **em** wyznacza wielkość definiowanego elementu zależnie od wielkości czcionki tego elementu. Jednostka **rem** zależy od wielkości podstawowej czcionki dokumentu. Jeżeli definiuje się wysokość wiersza lub odstępy między elementami za pomocą **em**, wraz ze zmianą wielkości tekstu wszystkie te miary zostaną proporcjonalnie dostosowane. Jest to szczególnie wygodne, gdy projektuje się strony na urządzenia mobilne.

W taki sam sposób można ustawić grubości obramowania nagłówka. Warto zaznaczyć, że choć na poniższym rysunku grubość obu nagłówków zdefiniowano na **1em**, to **h1** ma grubszą obwódkę niż **h3**, ponieważ czcionka w **h1** jest większa niż **h3**.

```
1. h1 {border: 1em solid #0000ff;}
2. h3 {border: 1em solid #0000ff;}
```




**Rys. 2.** Reguły nagłówków

### Ćwiczenie 1. Reguły zewnętrzne

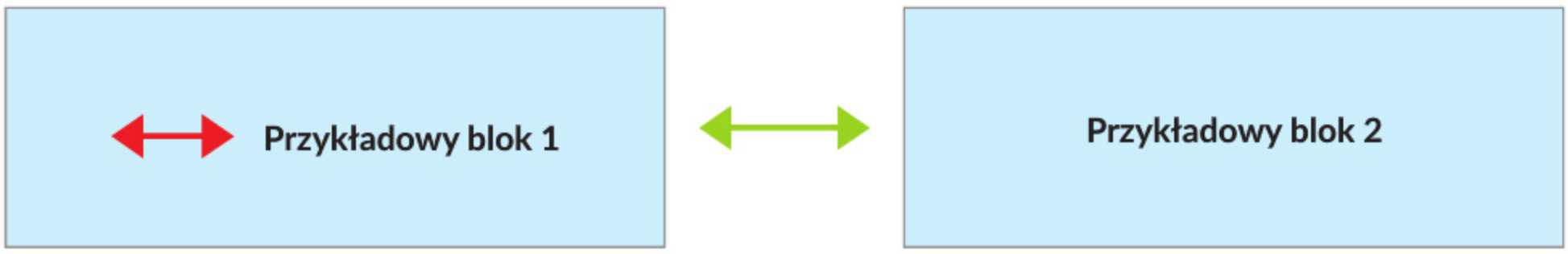
W dokumencie **index.html** strony SKW dodaj adres pliku stylu.

**ZAPAMIĘTAJ!**

**Kaskadowość stylów** oznacza hierarchię reguł. O ile nie określono inaczej, selektory, które znajdują się wewnątrz innych selektorów, dziedziczą wartości właściwości tych selektorów, np. kolor zdefiniowany dla elementu **<body>** ma zastosowanie również w elemencie **<p>**. Jeśli dwie reguły mają taką samą wagę, to stosowana jest ostatnia.

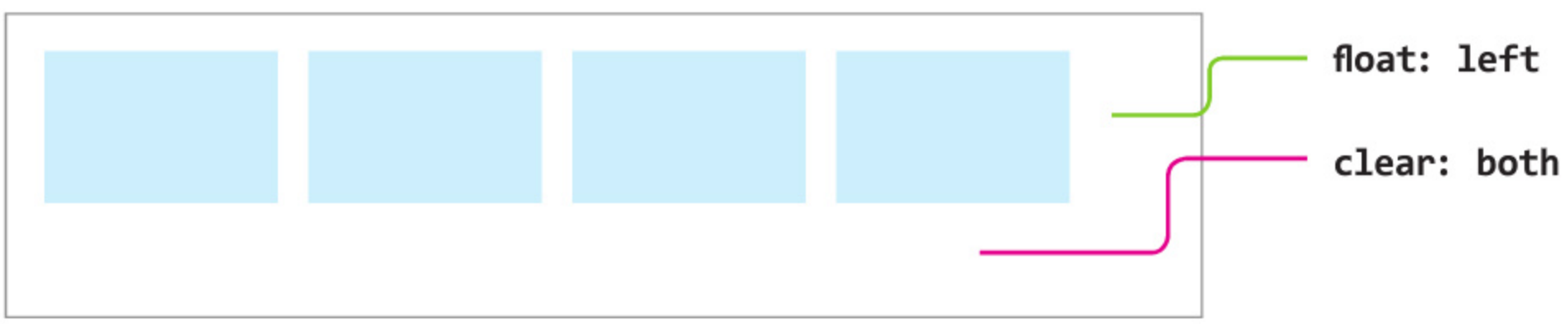
## MARGINESY I OBLEWANIE ELEMENTÓW

Podczas opisywania kolejnych elementów strony można określić właściwości **padding** (przestrzeń wokół danego elementu, tj. margines wewnętrzny między obramowaniem a zawartością obiektu) oraz **margin** (przestrzeń między elementami, tj. margines zewnętrzny między obramowaniem a sąsiadującymi obiektami).




**Rys. 3.** Różnica między odstępami **padding** (czerwona strzałka) i **margin** (zielona strzałka)

Kolejną ważną właściwością jest **float**. Określenie tej właściwości sprawia, że dany element jest przeciągany na lewo (**left**) bądź na prawo (**right**), a reszta elementów go otacza. Jeżeli kilka elementów obok siebie będzie miało właściwość **float**, ułożą się one obok siebie, dlatego ostatni musi mieć ustawioną właściwość **clear: both**.




**Rys. 4.** Elementy przyciągane do lewej wraz z zakończeniem

## CZCIONKI

Nie na wszystkich komputerach oraz urządzeniach mobilnych zestaw dostępnych czcionek jest ten sam. Jak to zrobić, żeby tekst zawsze wyświetlał się poprawnie? Można zastosować standardową czcionkę lub osadzić wybraną czcionkę na stronie. Warto skorzystać z bezpłatnych czcionek Google'a (**https://fonts.google.com**). Na stronie pojawi się instrukcja, jak dołączyć wybraną czcionkę. Potem wystarczy wpisać podany kod

HTML do pliku **index.html** przed deklaracją dołączenia stylów do strony, a podany kod CSS – jako regułę dla nagłówka. Dzięki temu czcionka nie będzie instalowana na komputerze użytkownika – będzie dynamicznie dołączana do strony.

## Ćwiczenie 2. Wygląd nagłówka i stopki

Zdefiniuj style nagłówka i stopki serwisu SKW (wzór poniżej). Logo umieść w lewym górnym rogu, obok nazwę koła, a po prawej stronie – linki. Stopka powinna zajmować całą szerokość strony, z wyjątkiem marginesów.




1. Utwórz nowy plik tekstowy **style.css**.
2. Zdefiniuj kolejne style nagłówka: kolor, czcionkę, marginesy zewnętrzne i wewnętrzne, odstępy między literami i ramkę.

- Aby uzyskać efekt zaokrąglonych rogów, dodaj obwódkę (**border**) i właściwość **border-radius**.

```
1. header {
2.     color: #313daf;
3.     font-family: 'Roboto Condensed', sans-serif;
4.     margin: 1em;
5.     padding: 1em;
6.     letter-spacing: 0.1em;
7.     border: 3px solid #313daf;
8.     border-radius: 15px;
9. }
```

- W przypadku bloku zamykającego z klasą **koniec** zastosuj zapis **clear: both**, aby zakończyć opływanie (następny blok nie będzie już układać się obok, ale poniżej).

```
1. .koniec {
2.     clear: both;
3. }
```

3. Zdefiniuj kolejne style stopki: kolor tła, marginesy zewnętrzne i wewnętrzne, stopień czcionki oraz wysokość wiersza.

```
1. footer {
2.     background: #cccccc;
3.     margin: 1em;
4.     padding: 1em;
5.     font-size: 0.8em;
6.     line-height: 1.5em;
7. }
```

3. Zapisz pliki i wyświetl stronę w przeglądarce.

## Ćwiczenie 3. Wygląd zasadniczej części strony

Dodaj style, które zdefiniują wygląd kart na stronie głównej SKW (wzór poniżej).




1. Otwórz pliki tekstowe **index.html** oraz **style.css**.

2. Sprawdź w pliku **index.html** klasę nadaną białemu obszarowi otoczonemu ramką, a następnie zdefiniuj ten obszar w pliku **style.css** – podaj grubość i kolor ramki oraz jej odległość od pozostałych elementów.

```
1. .karta {
2.     border: 1px solid #cccccc;
3.     margin-bottom: 1em;
4. }
```

3. Sprawdź w pliku **index.html** klasę nadaną prostokątnej winietce w prawym górnym rogu karty, a następnie zdefiniuj ten obszar w pliku **style.css**. Określ, w jaki sposób ma być przeciągany, a także ustaw marginesy, kolor tła oraz kolor czcionki.

```
1. .info {
2.     float: right;
3.     padding: 0.5em;
4.     margin: 0em;
5.     background: #313daf;
6.     color: #ffffff;
7. }
```

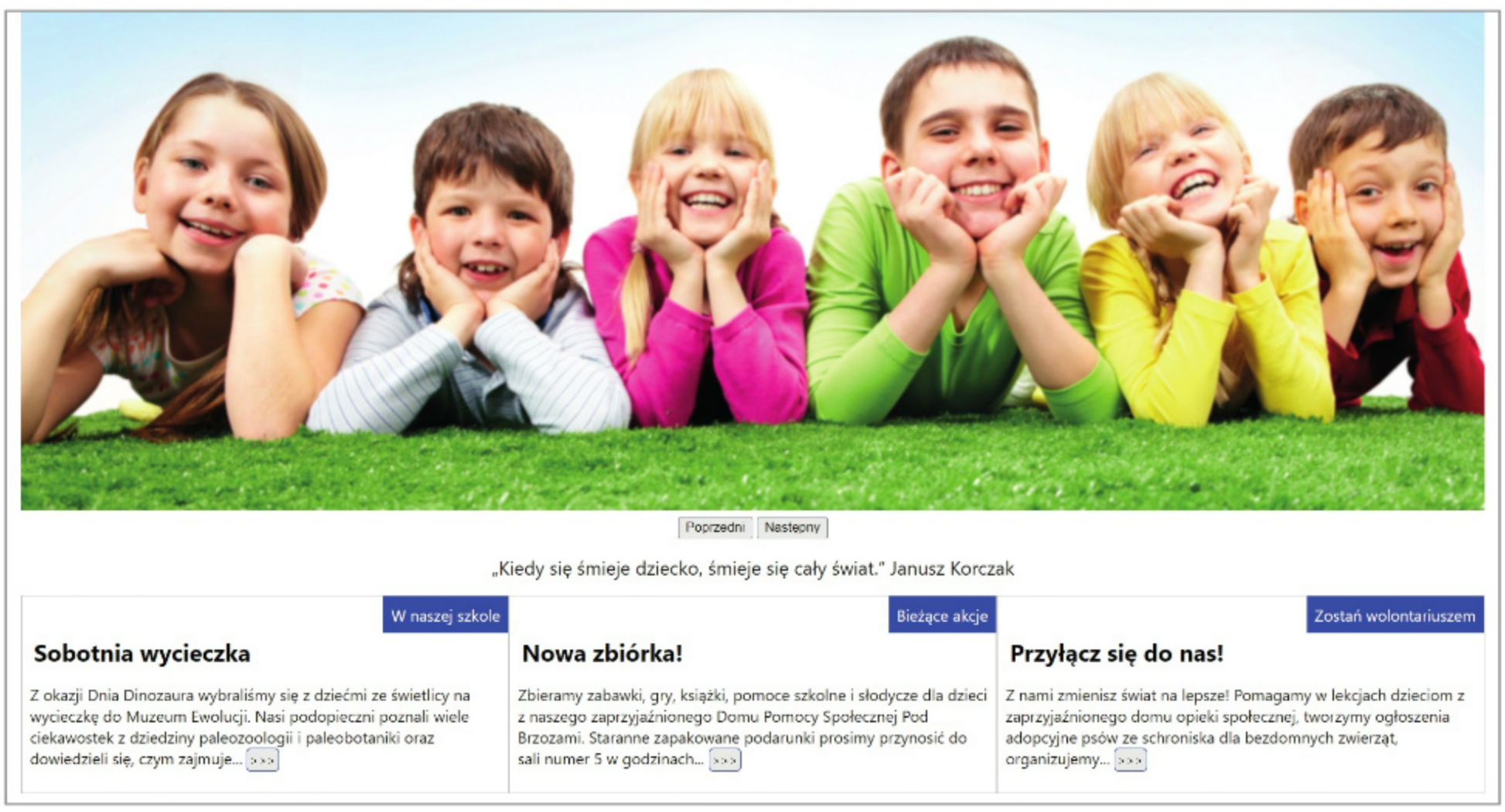
4. Sprawdź w pliku **index.html** klasę nadaną przyciskowi na karcie, a następnie zdefiniuj ten obszar w pliku **style.css**. Określ kolor, zwróć uwagę na zaokrąglenia i zmniejszenie czcionki. Przykładowy kod:

```
.przycisk {
    border: 1px solid #313daf;
    font-size: 0.8em;
    padding: 0.2em;
    border-radius: 5px;
    background: #eeeeee;
}
```

5. Zapisz pliki i wyświetl stronę w przeglądarce.

## Ćwiczenie 4. Grafika responsywna

Zdefiniuj regułę dotyczącą zdjęcia na stronie głównej SKW tak, aby obrazek zawsze zajmował całą szerokość strony, z wyjątkiem marginesów.



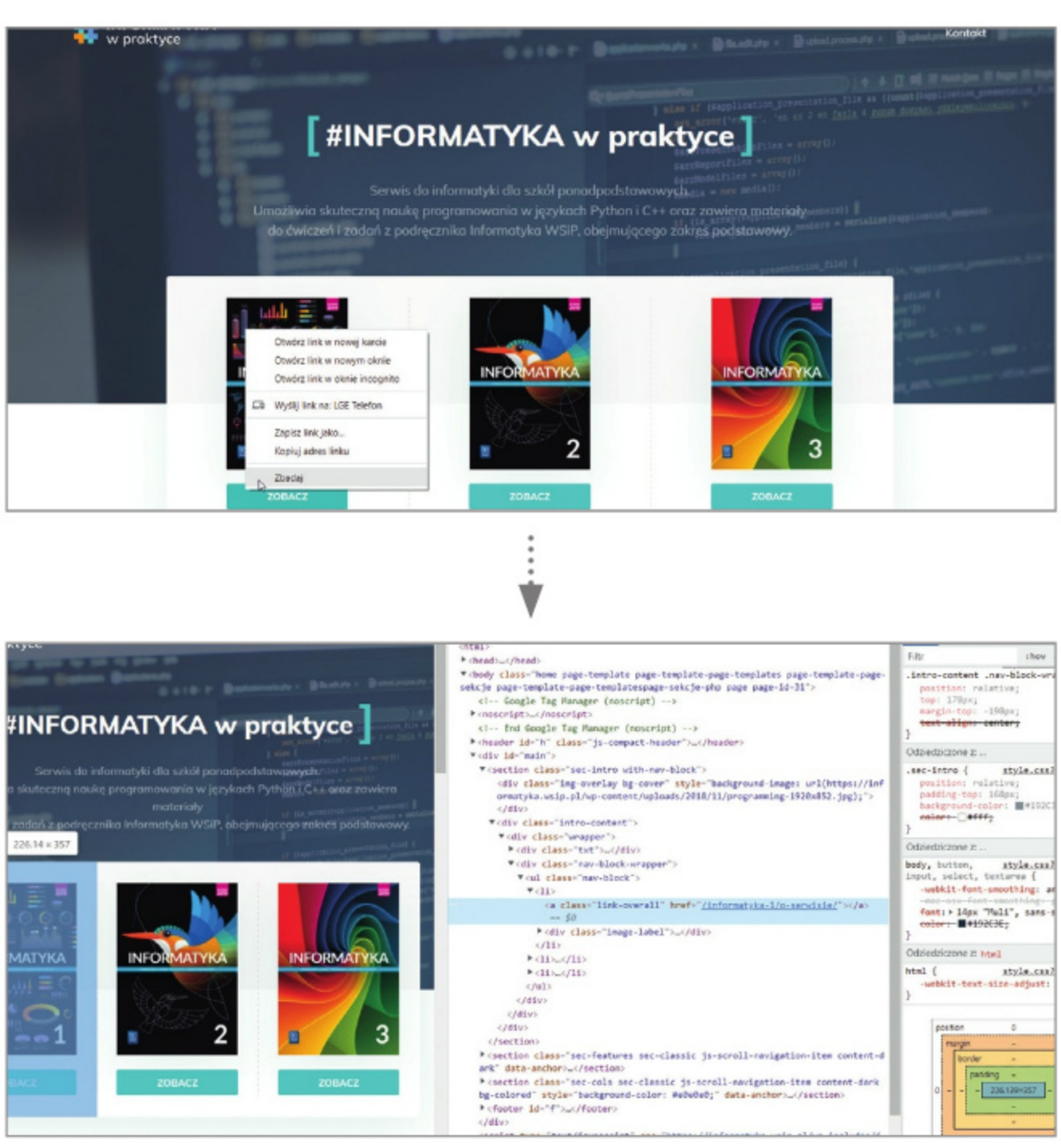

1. Sprawdź w pliku **index.html** klasę nadaną obiektom znajdującym się w rzędzie, a następnie zdefiniuj ten obszar w pliku **style.css** zgodnie z poleceniem.

- Odnieś się do klasy i znacznika.
- Właściwość **width** ustaw na **100%**, właściwość **height** ustaw tak, aby była dobierana proporcjonalnie do szerokości (**auto**) – dzięki temu zdjęcia nie będą zniekształcone.

```
.row img {
    width: 100%;
    height: auto;
}
```

**W PRAKTYCE**

Dodawania wybranych elementów do strony można nauczyć się przez analizę opublikowanych stron. Wystarczy wejść na daną stronę i wskazać kursorem myszy określony element, a następnie z menu podręcznego wybrać – w zależności od przeglądarki – **Zbadaj** (Chrome), **Zbadaj element** (Firefox) lub **Sprawdź element** (Edge). Wyświetlą się wtedy odpowiedzialne za niego kody HTML i CSS. Poniżej przedstawiono badanie przycisku z linkiem w przeglądarce Chrome.




## UKŁAD KART

Reguły **@media** umożliwiają definiowanie różnych wartości stylów w zależności od medium, na jakim wyświetlana jest strona (również w zależności od szerokości monitora). Zmieniać można w zasadzie wszystkie własności elementów, ale najczęściej używane są własności służące do ułożenia tych elementów, tj. **width**, **height**, **display**, **float**, **background**, oraz wielkość czcionki.

```
1. @media (min-width: 480px) and (max-width: 800px) {
2.    div {
3.        background-color: blue;
4.    }
5. }
```

**Rys. 5.** Przykładowa reguła: jeśli szerokość okna przeglądarki będzie większa niż 480 pikseli i mniejsza niż 800 pikseli, to dla danego elementu **div** tło będzie miało kolor niebieski

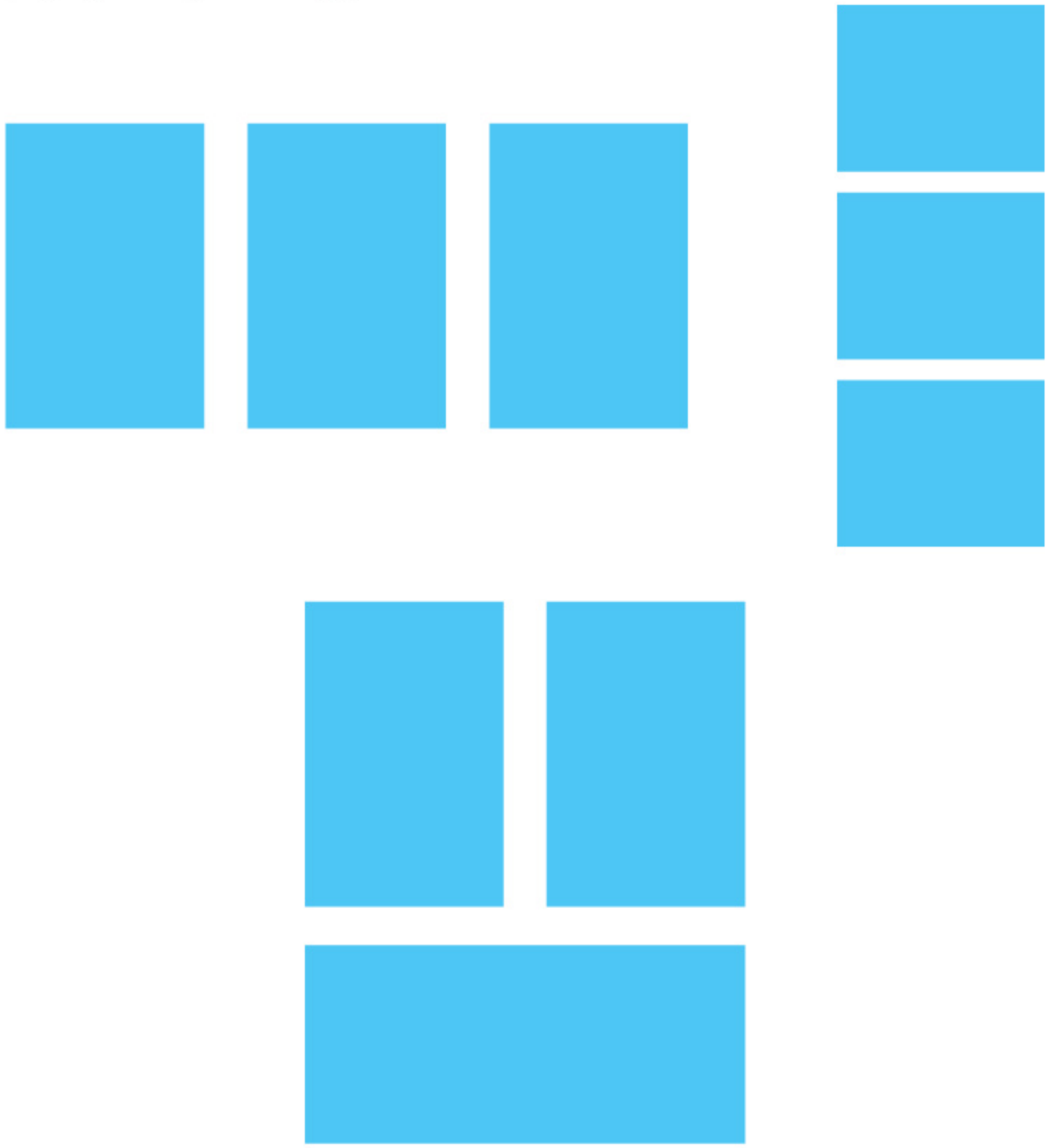
Aby określić pożądany układ elementów na stronie w zależności od szerokości ekranu, należy opisać odpowiednio reguły w pliku stylów, tj. określić warunek i oczekiwaną reakcję. Warto zauważyć, że wprowadzenie reguł może wiązać się z koniecznością dodania klas (i uzupełnienia pliku HTML). Chociaż nazwy klas mogą być dowolne, należy stosować przejrzyste konwencje nazewnicze. Zwyczajowo przyjmuje się następujące nazwy klas:

- **`.col-xs-*`** – smartfony;
- **`.col-sm-*`** – smartfony w pozycji horyzontalnej;
- **`.col-md-*`** – tablety;
- **`.col-lg-*`** – tablety w pozycji horyzontalnej, laptopy.

Podczas projektowania strony często dzieli się ją umownie na siatkę 12-elementową i określa szerokość elementów przez liczbę kolumn, które zajmują, np. 12 oznacza całą szerokość (100 %), 6 – połowę (50 %), a 4 – jedną trzecią (ok. 33,33 %).

## Ćwiczenie 5. Układ kart

Zdefiniuj trzy reguły, które sprawią, że na szerokich ekranach (992 px i więcej) karty będą wyświetlane jedna obok drugiej, na średniej wielkości ekranach (768–991 px) – pierwsze dwie w jednym rzędzie, a ostatnia w drugim, a na małych ekranach (do 767 px) – jedna pod drugą.



1. Zdefiniuj reguły w w pliku **style.css** zgodnie z poleceniem.

- Podczas dodawania klas zastosuj przejrzyste konwencje nazewnicze, np. zaszyj w nazwie klasy jej szerokość.

```
@media (min-width: 992px) {
   .col-lg-4 {width: 33.33%;}
   .col-lg-6 {width: 50%;}
}
@media (min-width: 768px) and (max-width: 991px) {
   .col-md-6 {width: 50%;}
   .col-md-12 {width: 100%;}
}
@media (max-width: 767px) {
   .col-sm-12 {width: 100%;}
}
```

2. Zmodyfikuj plik HTML przez dodanie odpowiednich klas.

```
<div class="col-lg-4 col-md-6 col-sm-12">
    <div class="karta">
        <div class="info">W naszej szkole</div>
        <h2>Sobotnia wycieczka</h2>
        <p>Z okazji Dnia Dinozaura… </p>
    </div>
</div>

<div class="col-lg-4 col-md-6 col-sm-12">
    <div class="karta">
        <div class="info">Bieżące akcje</div>
        <h2>Nowa zbiórka!</h2>
        <p>Zbieramy zabawki, gry… </p>
    </div>
</div>

<div class="col-lg-4 col-md-12 col-sm-12">
    <div class="karta">
        <div class="info">Zostań wolontariuszem</div>
        <h2>Przyłącz się do nas!</h2>
        <p>Z nami zmienisz…</p>
    </div>
</div>
<div class="koniec"></div>
```

### Ćwiczenie 6. Pozostałe części serwisu

Zdefiniuj wygląd elementów ze stron przygotowanych podczas realizacji ćwiczenia 4 z lekcji 22. Sprawdź, czy zdefiniowany arkusz stylów działa też na dodanych stronach.

## PODSUMOWANIE

- HTML pozwala na zdefiniowanie struktury i zawartości dokumentu, CSS odpowiada za jego sformatowanie.
- Reguły opisujące style tworzą hierarchię. Selektory, które znajdują się wewnątrz innych selektorów, dziedziczą wartości właściwości tych selektorów. Jeśli dwie reguły mają taką samą wagę, to stosowana jest ostatnia.
- Każda reguła określająca styl dla danego elementu dokumentu składa się z selektora oraz deklaracji otoczonej nawiasami klamrowymi. Selektorem może być znacznik języka HTML, jego niepowtarzalna nazwa (**`id`**) czy klasa elementu (**`class`**). Deklaracje określają formatowanie i składają się z nazwy właściwości oraz jej wartości.
- Reguły wspomagają projektowanie stron na różne urządzenia, gdyż umożliwiają definiowanie wartości stylów w zależności od medium, na jakim wyświetlana jest strona, np. w zależności od szerokości monitora.

## PYTANIA SPRAWDZAJĄCE

1. Jak działają reguły CSS? Która z nich jest ważniejsza, a która mniej ważna?
2. Jak za pomocą reguł określić wygląd pojedynczego elementu – akapitu, nagłówka, przycisku?
3. Jak zdefiniować reguły, aby dostosować wygląd strony do wyświetlania na różnych urządzeniach?

## PYTANIA QUIZOWE

1. Jaki kolor obwódki będzie miał element klasy **`karta`**?

```
.karta {
  border: 1px solid #ff0000;
  margin: 1em;
}
```

**a.** Czarny.
**b.** Czerwony.
**c.** Zielony.
**d.** Niebieski.

2. Jaka właściwość określa szerokość zaznaczoną strzałkami?




   a. `margin`
   b. `padding`
   c. `H1`
   d. `letter-spacing`

3. Które ze stwierdzeń na temat reguł `@media` jest prawdziwe?
   a. Opisują sposób wyświetlania grafiki, a nie tekstu.
   b. Opisują sposób wyświetlania tekstu, a nie grafiki.
   c. Definiują reguły w zależności od szerokości monitora.
   d. Likwidują kaskadowość.

## ZADANIA DODATKOWE

### Zadanie 1. Układ kart

Zapisz regułę na czterech kartach informacyjnych rozmieszczonych w następujący sposób: na szerokim ekranie 4, na średnim 2 + 2, na wąskim pojedynczo.

### Zadanie 2. Dodawanie animacji

Dzięki takim grupom właściwości CSS, jak **`transition`**, **`animation`**, **`transform`**, można dodawać animacje na stronie. Dowiedz się więcej na ten temat i wstaw elementy animacji na swoją stronę.

### Zadanie 3. Różna wielkość tekstu

Zmodyfikuj reguły CSS w taki sposób, żeby wyświetlany tekst był różnej wielkości w zależności od urządzenia: na szerokim ekranie – większy, a na małym – odpowiednio mniejszy. Przetestuj stronę na różnych urządzeniach.



# 24. Elementy dynamiczne

- Wprowadzenie do języka JavaScript
- Projektowanie elementów dynamicznych
- Pisanie skryptów w języku JavaScript

## ZADANIE NA START

### Jak to rozumiesz

Poniższa funkcja **test** została zdefiniowana w języku JavaScript. Przeanalizuj kod i podaj wyniki funkcji dla parametrów 2 i 7.

```
1. function test(n) {
2.     var liczba = 0;
3.     for (var i = 1; i < n; i++) {
4.         liczba = liczba + 1;
5.     }
6.     return liczba;
7. }
```



## ABC JĘZYKA JAVASCRIPT

JavaScript to język programowania, który po dodaniu do dokumentu HTML wprowadza interaktywność. Jest niezwykle wszechstronny – można w nim napisać proste skrypty przeglądania obrazków w wyświetlanej galerii czy zmiany napisów w zależności od dnia tygodnia, jak również gry, animowane grafiki 2D i 3D oraz aplikacje oparte na bazach danych.

Skrypty JavaScript można umieszczać w dokumencie HTML albo w plikach zewnętrznych. Drugie rozwiązanie gwarantuje większą przejrzystość kodu (kody HTML i JavaScript są oddzielone, co ułatwia analizę i dokonywanie wszelkich poprawek), a poza tym pozwala na wykorzystanie tego samego kodu na różnych stronach danego serwisu, więc przyśpiesza ładowanie strony w przeglądarce. Z reguły pliki JavaScript mają rozszerzenie .js. Jako atrybut **src** podaje się adres pliku wraz z jego rozszerzeniem.

```
1. <script src="foto.js"></script>
```

**Rys. 1.** Dołączenie pliku JS zawierającego skrypt

## GALERIA Z PRZYCISKAMI

Aby utworzyć galerię zdjęć obsługiwaną za pomocą przycisków, np. << i >>, należy najpierw umieścić w odpowiednim miejscu w dokumencie HTML kod odpowiedzialny za dodanie przycisków.



**Rys. 2.** Przykładowe przyciski umożliwiające interakcję internauty z zawartością strony

Zdarzenie **onclick** powoduje, że wywoływana jest odpowiednia, wcześniej zdefiniowana funkcja, np. **poprzedni()**.

```
1.  <button onclick="poprzedni()"><<</button>
2.  <button onclick="nastepny()">>></button>
```

**Rys. 3.** Dodawanie przycisków

Można zadeklarować dwie zmienne lokalne. Zmienne deklaruje się, zaczynając od słowa kluczowego **var**. Zmienna **obrazki** będzie służyć do przechowywania nazw plików graficznych, które mają się pokazywać na danej stronie. Wykorzystanie zmiennej tablicowej sprawia, że łatwiejsze staje się zapamiętanie wielu nazw plików. Zmienna **nr** – będąca liczbą całkowitą – będzie przechowywać numer aktualnie wyświetlanego obrazka.

```
1. var obrazki = ["zosia.png", "adam.png", "basia.png"];
2. var nr = 0;
```

**Rys. 4.** Deklaracja i inicjalizacja zmiennych

Kliknięcie przycisku >> ma wywołać funkcję **nastepny()**, która zmienia obraz na kolejny. Należy w tym celu wykorzystać metodę **getElementById**. Dzięki unikatowej nazwie elementu (**id**) można się do niego odwoływać i go zmieniać.

```
1. function nastepny() {
2.    var galeria = document.getElementById("galeria");
3.    nr++;
4.    if (nr >= obrazki.length) {
5.       nr = 0;
6.    }
7.    galeria.src = "grafika/" + obrazki[nr];
8. }
```

**Rys. 5.** Kod funkcji **nastepny()**

Kliknięcie przycisku << ma wywołać funkcję **poprzedni()**, która zmienia obraz na poprzedni. Oznacza to, że trzeba zmniejszyć wartość zmiennej **nr**. Po wyświetleniu

pierwszego obrazka (o numerze 0) należy wrócić do wyświetlania ostatniego obrazka (o numerze: długość tablicy –1).

```
1. function poprzedni() {
2.   var galeria = document.getElementById("galeria");
3.   nr--;
4.   if (nr < 0) {
5.     nr = obrazki.length - 1;
6.   }
7.   galeria.src = "grafika/" + obrazki[nr];
8. }
```

**Rys. 6.** Kod funkcji **poprzedni()**

## Ćwiczenie 1. Tworzenie galerii obrazków z przyciskami

Zastąp zdjęcie na stronie głównej SKW galerią zdjęć – fotografie powinny się zmieniać po naciśnięciu przycisków **Następny** i **Poprzedni**. Skrypt zapisz w oddzielnym pliku.

1. W pliku HTML zaktualizowanym podczas poprzedniej lekcji dodaj **id** obrazka, do którego będziesz się odwoływać w JavaScripcie.

```
1. <div class="row">
2.       <img id="galeria" src="grafika/dzieci.png" alt="Uśmiechnięte
         dzieci">
3. </div>
```

2. Pod zdjęciem umieść kod odpowiedzialny za dodanie i wyśrodkowanie przycisków.

```
1. <div class="row">
2.     <img id="galeria" src="grafika/dzieci.png" alt="Uśmiechnięte dzieci">
3.     <center><button onclick="poprzedni()">Poprzedni</button>
4.     <button onclick="nastepny()">Następny</button><center>
5.   </div>
```

3. W sekcji **<head>** umieść nazwę skryptu obsługującego przyciski.

```
1. <script src="galeria.js"></script>
```

4. Zdefiniuj plik JavaScript.

- Otwórz nowy plik i zapisz go jako **galeria.js**.
- Zadeklaruj zmienną typu tablica o nazwie **obrazki** do przechowywania nazw plików graficznych, które będziesz trzymać w folderze **grafika**, oraz zmienną **nr** przechowującą numer wyświetlanego obrazka.

```
1. var obrazki = ["dzieci_1.png", "dzieci_2.png", "dzieci_3.png"];
2. var nr = 0;
```

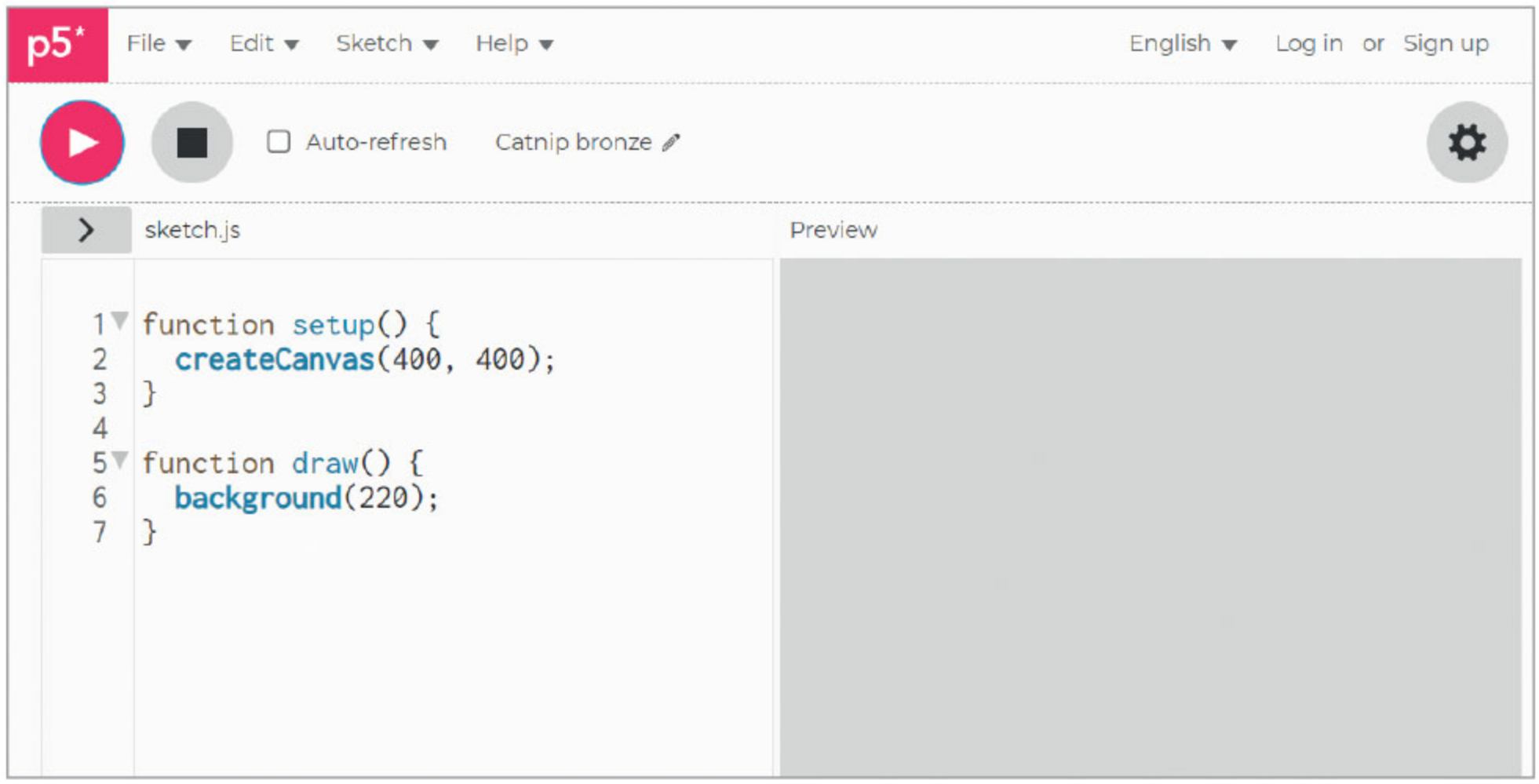
5. Zdefiniuj funkcje **nastepny()** i **poprzedni()**.

```
1. function nastepny() {
2.   var galeria = document.getElementById("galeria");
3.   nr++;
4.   if (nr >= obrazki.length) {
5.     nr = 0;
6.   }
7.   galeria.src = "grafika/" + obrazki[nr];
8. }
9. function poprzedni() {
10.  var galeria = document.getElementById("galeria");
11.  nr--;
12.  if (nr < 0) {
13.    nr = obrazki.length - 1;
14.  }
15.  galeria.src = "grafika/" + obrazki[nr];
16.}
```

6. Zapisz plik i wyświetl stronę w przeglądarce.

## BIBLIOTEKA P5.JS

Podczas tworzenia stron WWW można skorzystać z biblioteki p5.js, która pozwala manipulować obiektami HTML5, takimi jak tekst, dane wejściowe, obraz czy dźwięk. Edytor online **https://editor.p5js.org** umożliwia pisanie i uruchamianie projektów multimedialnych.



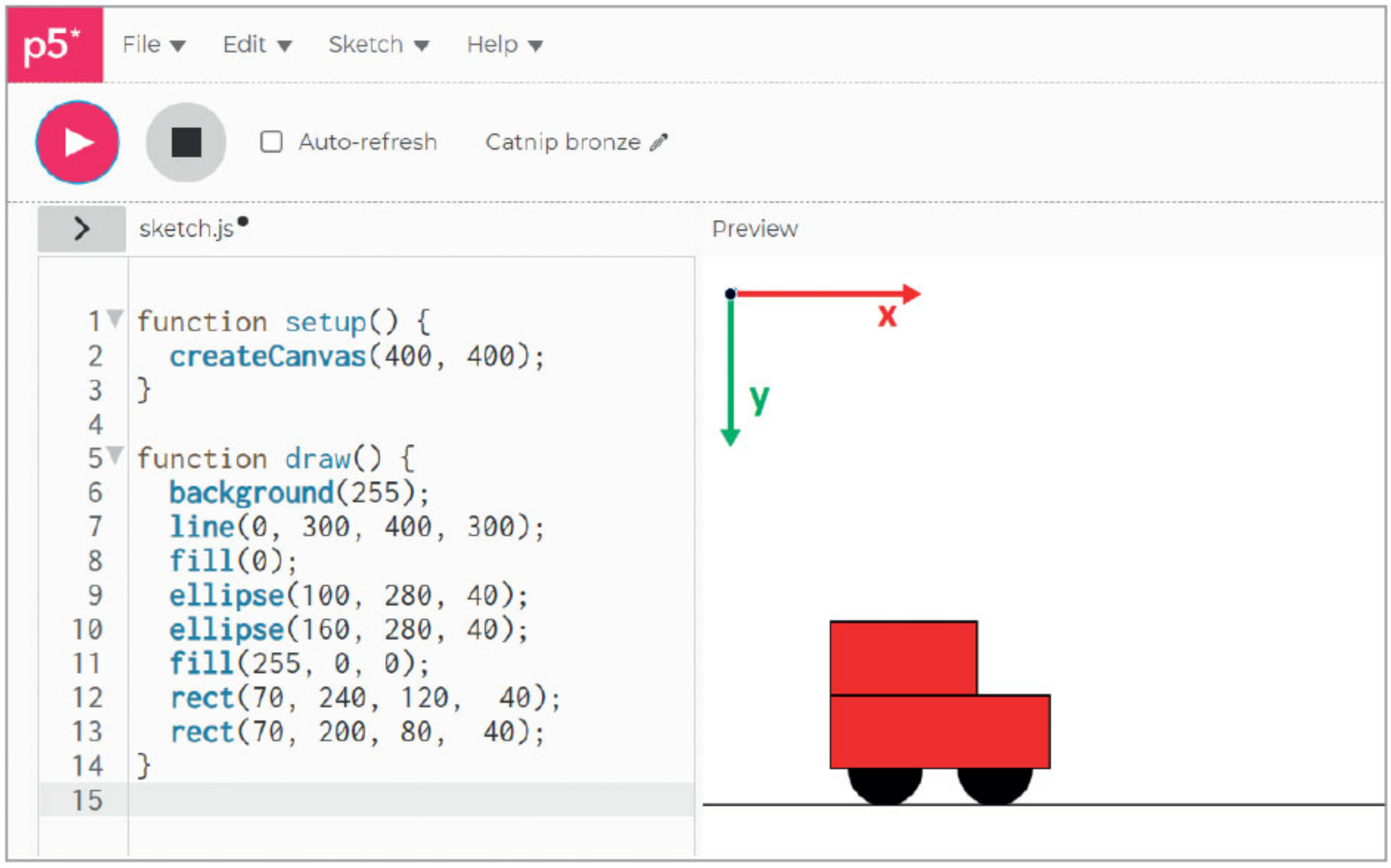

**Rys. 7.** Edytor online biblioteki p5

Każdy projekt rozpoczyna się gotowymi funkcjami: **setup() i draw()**. W funkcji **setup()** umieszcza się polecenia wykonywane raz, np. **createCanvas(400, 400)**, które

tworzy płótno o rozmiarze 400 × 400. W funkcji **draw()** umieszcza się polecenia wykonywane w pętli, np. **background(220)**, czyli wypełnianie płótna szarym tłem.

Do rysowania można wykorzystać następujące funkcje:

- **line(x1, y1, x2, y2)** – rysuje linie od punktu *x1*, *y1* do punktu *x2*, *y2*;
- **ellipse(x, y, s)** – rysuje okrąg o środku w punkcie *x*, *y* i średnicy *s*;
- **rect(x, y, szer, wys)** – rysuje prostokąt o lewym dolnym rogu *x*, *y* i podanej szerokości oraz wysokości;
- **fill(r, g, b)** – ustawia kolor malowania na podany.



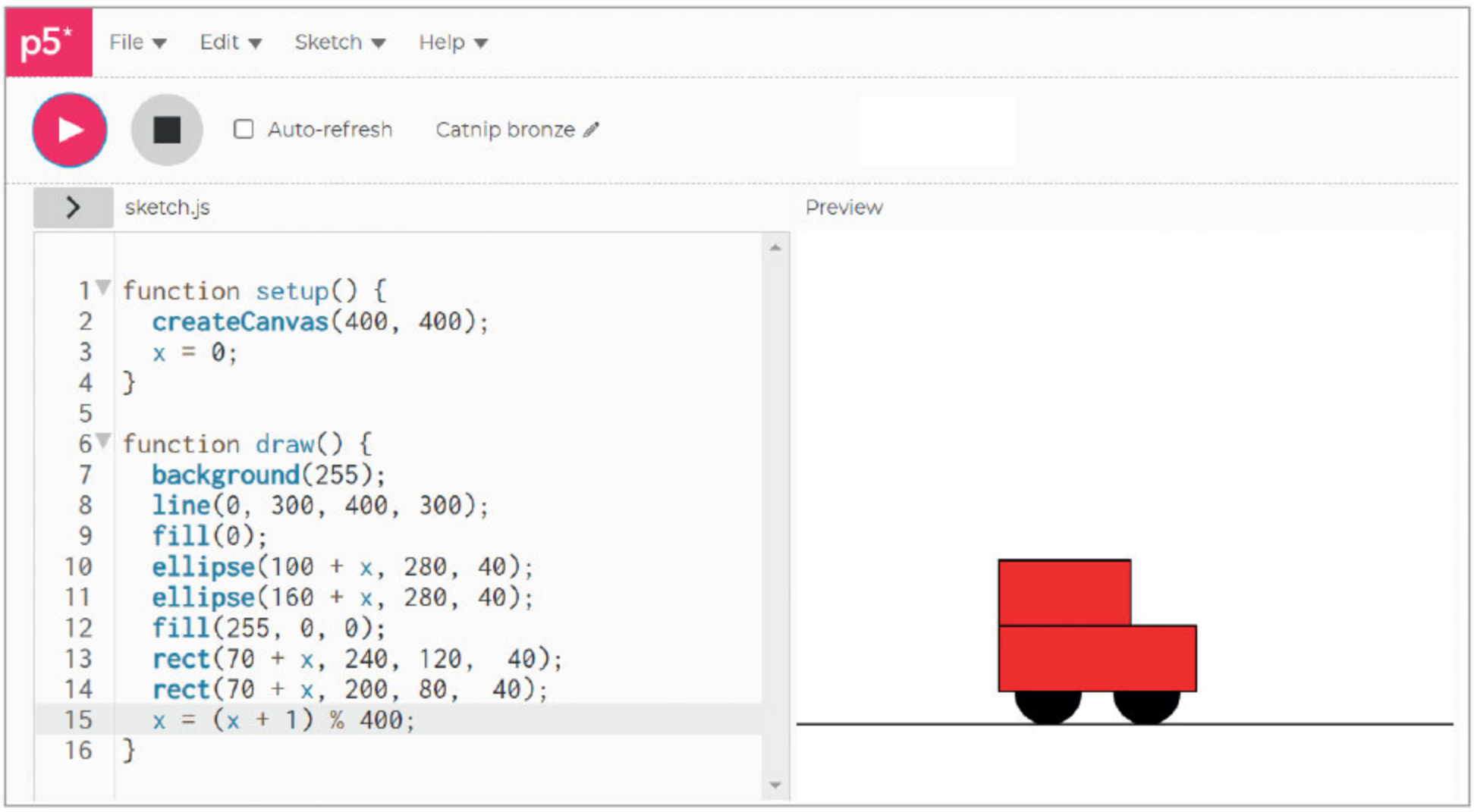

**Rys. 8.** Przykładowy kod odpowiadający za narysowanie samochodu. Początek układu współrzędnych znajduje się w lewym górnym rogu, przy czym x rośnie w prawo, a y – w dół

Aby dodać projekt do witryny, należy w sekcji **head** pliku HTML dołączyć kod biblioteki p5.js oraz utworzony skrypt, na przykład **sketch.js**.

```
1. <html>
2.   <head>
3.     <script src="https://cdnjs.cloudflare.com/ajax/libs/p5.js/1.4.0/
   p5.js"></script>
4.     <script src="sketch.js"></script>
5.   </head>
6.   <body>
7.
8.   </body>
9. </html>
```

**Rys. 9.** Dołączanie projektu p5.js do strony WWW

Ponieważ funkcja **`draw()`** jest wywoływana 60 razy na sekundę, do stworzenia animacji wystarczy sparametryzować rysunek tak, aby współrzędne zależały od zmiennej. Poniżej dodano zmienną *x*, która powoduje, że samochód będzie poruszał się poziomo. W funkcji **`setup()`** trzeba ustalić wartość początkową zmiennej, a potem zmienić jej wartość.




**Rys. 10.** Zmodyfikowany kod odpowiadający za narysowanie i poruszanie się samochodu

### Ćwiczenie 2. Uśmiechnięte słońce

Narysuj uśmiechnięte słońce z wykorzystaniem biblioteki p5.js. Dodaj je do witryny SKW.

### Ćwiczenie 3. Animowane słońce

Zanimuj słońce z ćwiczenia 2 – dodaj ruch oczu. Dodaj je do witryny SKW.



## PODSUMOWANIE

- JavaScript to bardzo praktyczny język, za pomocą którego można projektować elementy dynamiczne – przygotować galerię zdjęć, dodać zmieniający się tekst czy też zdefiniować quiz.

- Do konkretnych elementów określonych przez **`id`** należy się odwoływać poleceniem **`document.getElementById(id_elementu)`**. W skrypcie można analizować jego właściwości, a nawet je zmieniać.
- Elementy interaktywne na stronie mogą być wywoływane kliknięciem (zdarzenie **`onclick`**). Warto zdefiniować odpowiednią funkcję, odpowiedzialną za wykonanie interakcji.
- Biblioteka JavaScript p5.js umożliwia manipulowanie obiektami HTML5 w postaci tekstu, obrazów, dźwięków itd.

## PYTANIA SPRAWDZAJĄCE

1. Gdzie umieszcza się skrypty JavaScriptu – w kodzie HTML czy w osobnych plikach?
2. Jakie są podstawowe konstrukcje języka JavaScript?
3. W jaki sposób można wykorzystywać język JavaScript na stronach WWW?

## PYTANIA QUIZOWE

1. Co jest skutkiem zdarzenia **`onclick`**?
   **a.** Wywołanie wcześniej zdefiniowanej funkcji.
   **b.** Wywołanie następnej zdefiniowanej funkcji.
   **c.** Płynna zmiana obrazów.
   **d.** Animowanie za pomocą ruchów kursorem.
2. Które ze stwierdzeń na temat biblioteki JavaScript p5.js jest prawdziwe?
   **a.** Zastępuje HTML podczas tworzenia stron WWW.
   **b.** Ułatwia programowanie grafiki i elementów interaktywnych na stronie WWW.
   **c.** Waliduje stronę WWW.
   **d.** Ułatwia przesyłanie pliku **`index.html`** na serwer.
3. Jak uruchamia się funkcja **`setup()`**?
   **a.** Raz na początku.
   **b.** Raz na sekundę.
   **c.** 60 razy na sekundę.
   **d.** Raz przy zamykaniu strony.

## ZADANIE DODATKOWE

### Zadanie 1. Galeria z płynną zmianą obrazów

Aby stworzyć galerię, w której zdjęcia będą się zmieniać po określonym czasie, należy wykorzystać rekurencję (wywołanie funkcji przez samą siebie) oraz zastosować metodę **`setTimeout`**, która ma dwa parametry – funkcję do wywołania oraz liczbę milisekund przed uruchomieniem funkcji podanej jako pierwszy parametr. Do pliku **`galeria.js`** dopisz funkcję, za pomocą której obrazki będą się zmieniać na stronie co określony czas.



# 25. Publikacja i ocena strony WWW

- Metody testowania strony
- Publikowanie strony na serwerze
- Ocena końcowa strony

## ZADANIE NA START

**Testowanie strony**

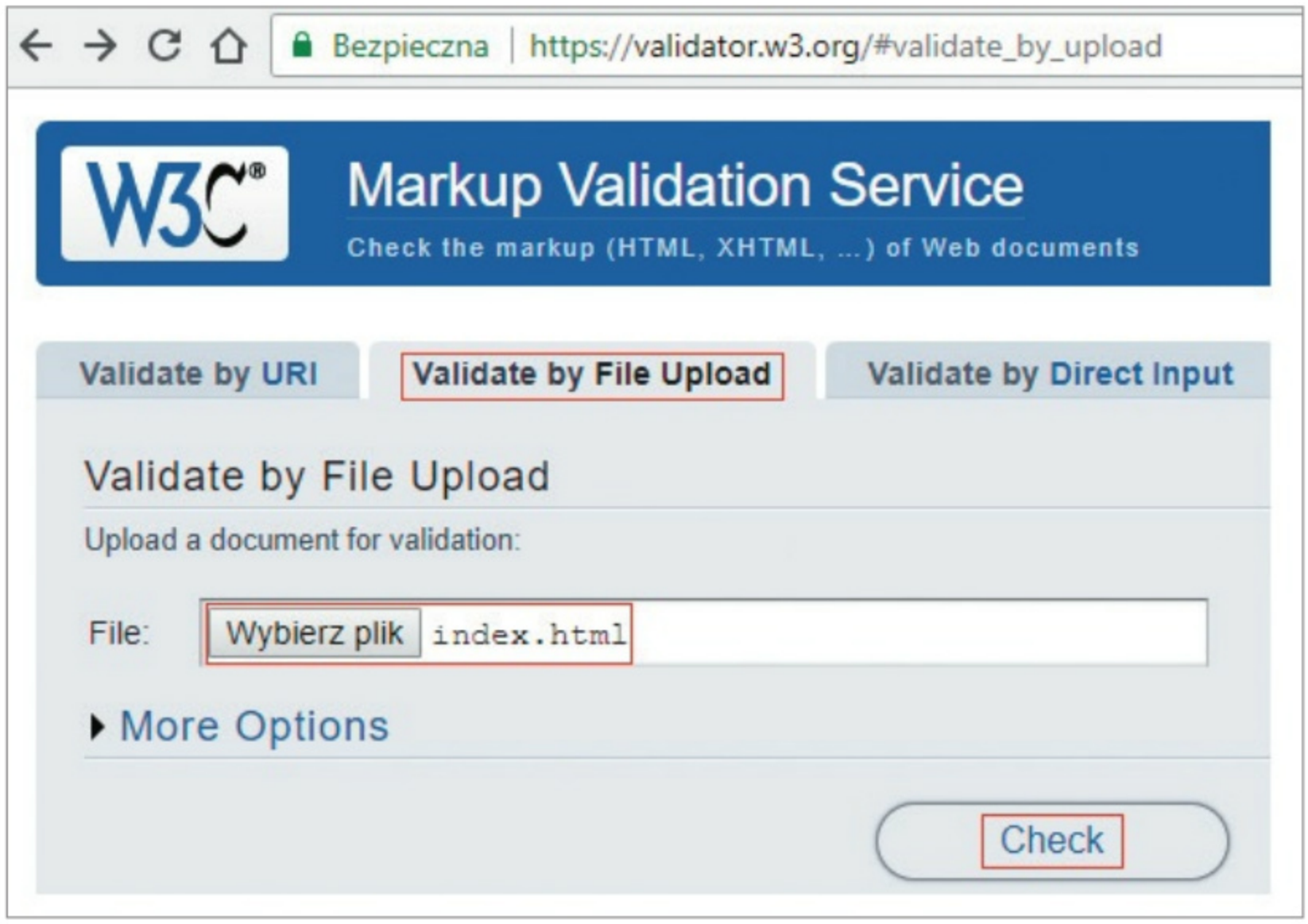
W trakcie weryfikacji i walidacji strony można wykryć wiele błędów i nieprawidłowości. Opracowana strona internetowa powinna przejść sporo testów, zanim zostanie opublikowana. Czy weryfikacja i walidacja programu komputerowego to ten sam proces? Scharakteryzuj krótko każdy z nich.



## SPRAWDZANIE POPRAWNOŚCI STRONY WWW

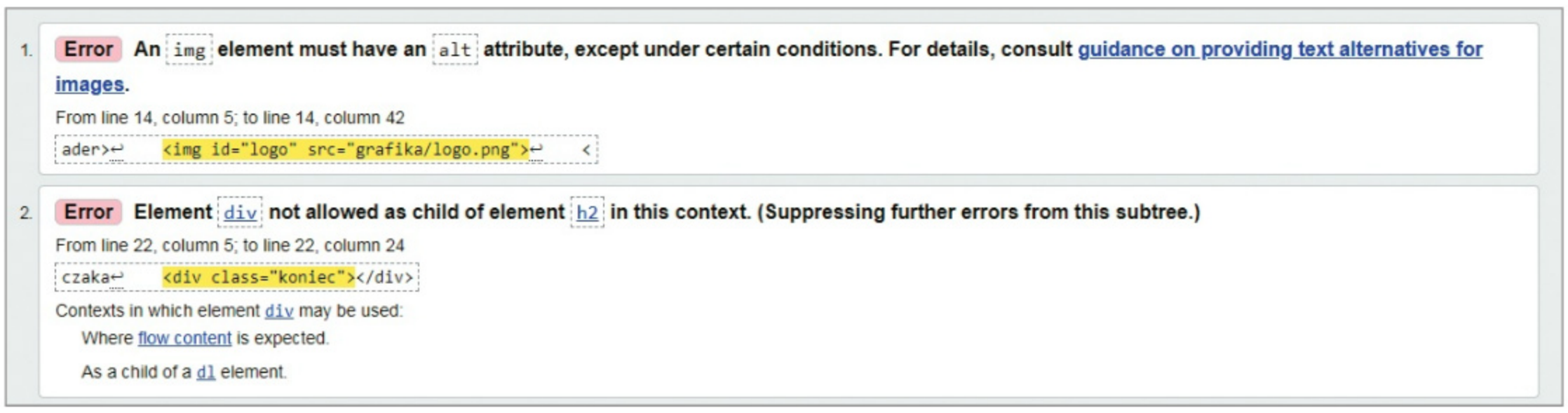
Integralnym elementem pracy nad stroną WWW jest testowanie i wprowadzanie poprawek, ponieważ strona nie powinna zawierać błędów, które wpływałyby na jej działanie. Przed opublikowaniem gotowej strony w sieci WWW należy ją solidnie przetestować. Dobrze jest też przeprowadzić automatyczną walidację, czyli sprawdzić zastosowane kody pod względem poprawności i obowiązujących standardów. Standardy stron WWW są wyznaczane przez The World Wide Web Consortium (W3C). W pierwszej kolejności weryfikuje się poprawność składniową dokumentu HTML, następnie waliduje się arkusz CSS.

Aby rozpocząć proces walidacji kodu HTML, w pasek adresu przeglądarki należy wpisać **https://validator.w3.org**, a następnie wybrać sposób walidacji, przesłać plik **index.html** do sprawdzenia i kliknąć **Check**. Strona, która nie zawiera błędów, powinna być właściwie wyświetlana we wszystkich przeglądarkach.




**Rys. 1.** Walidacja lokalnego pliku i jej efekt

W przypadku usterek walidator wskaże miejsca wystąpienia błędów i poda opis konkretnej nieprawidłowości. Na poniższym zrzucie widać, że w pierwszym przypadku nie dodano do obrazka atrybutu **alt**, a w drugim brakuje znacznika zamykającego dla **<h2>**.




**Rys. 2.** Walidacja strony z błędami

Przy walidacji arkusza CSS należy wpisać **https://jigsaw.w3.org/css-validator** w pasek adresu przeglądarki, a następnie wybrać sposób walidacji, przesłać plik do sprawdzenia i kliknąć **Sprawdź**.




**Rys. 3.** Wynik działania walidatora CSS

## Ćwiczenie 1. Sprawdzanie poprawności strony

Przeprowadź walidację strony według standardów W3C.

### WYBÓR SERWERA (FIRMY HOSTINGOWEJ)

Udostępnianiem stron internetowych w sieci zajmują się firmy hostingowe. Oprócz miejsca na serwerze na ogół dostaje się od nich również adres internetowy. Podczas rozpatrywania wyboru firmy hostingowej warto wziąć pod uwagę dwa kryteria:

- pojemność konta – jeśli strona składa się z tekstu i niewielkiej liczby zdjęć, nie trzeba sobie tym zawracać głowy, ale przy dużej liczbie wysokiej jakości zdjęć im większa pojemność, tym lepiej;
- transfer (rozmiar wszystkich plików i dokumentów pobieranych z całego serwisu podczas odwiedzin użytkowników). Przekroczenie limitu zazwyczaj skutkuje zablokowaniem witryny i wyświetleniem odpowiedniego komunikatu. Najczęściej limity dotyczą transferów miesięcznych, więc jeśli użytkownik przekroczy limit, jego strona zacznie się na nowo wyświetlać dopiero na początku następnego miesiąca. Dla przeciętnych stron miesięczny limit transferu o wielkości kilku gigabajtów powinien okazać się wystarczający.

**POSZUKAJ W INTERNECIE**

Poszukaj w sieci informacji na temat hostingowych serwisów internetowych przeznaczonych do publikowania stron.

## UMIESZCZANIE SERWISU NA SERWERZE

Po wybraniu serwisu hostingowego należy przeprowadzić rejestrację. Konieczne jest podanie poprawnego adresu e-mail, na który zostanie wysłana wiadomość z linkiem aktywacyjnym pozwalającym na zakończenie rejestracji. Warto zanotować najważniejsze dane umożliwiające dostęp do konta: login, hasło oraz adres serwera FTP (będą potrzebne podczas przesyłania strony na serwer). Na niektórych serwerach aktywacja konta nie następuje natychmiast – najpierw użytkownik dostaje e-mail, który o tym informuje.

Po aktywowaniu konta można przesłać pliki na serwer. Najprostsza metoda polega na wykorzystaniu przeglądarki, wpisaniu adresu serwera FTP i zalogowaniu się na swoje konto.

Można w tym celu wykorzystać również dowolny program będący klientem FTP.

Warto pamiętać, że niektóre serwery nie umożliwiają kopiowania plików przez program FTP. W takiej sytuacji trzeba zapoznać się z opisem przesyłania strony na danym serwerze i postępować według wskazówek.

Wszystkie pliki i foldery należy umieścić wprost w głównym katalogu konta, który wyświetlił się zaraz po zalogowaniu. Zazwyczaj jest to katalog o nazwie **public_html** lub **www**, ale może też mieć w nazwie login użytkownika. Jeśli połączenie jest wykonywane z komputera, do którego mają dostęp inne osoby, np. w szkolnej pracowni, należy zrezygnować z zapisywania hasła i usunąć połączenie z listy sesji.

**W PRAKTYCE**

Po wgraniu plików na serwer należy uruchomić stronę i sprawdzić wszystkie łącza, obrazki, strony itp. W każdej chwili można wykonać modyfikację serwisu. Poprawione pliki należy ponownie przesłać na serwer.

### Ćwiczenie 2. Publikowanie plików na serwerze

Prześlij pliki składające się na twoją witrynę na wybrany serwer i sprawdź, czy wszystko poprawnie działa. Przetestuj ręcznie i automatycznie – przeprowadź walidację. Porównaj efekt końcowy z założeniami. Czy udało ci się zrealizować cel?

## PODSUMOWANIE

- Na końcowy etap tworzenia strony składają się: testowanie, publikacja i ocena merytoryczna.
- Testowanie to ręczne sprawdzanie poprawności oraz automatyczna walidacja, czyli weryfikowanie poprawności składniowej dokumentu.

- Serwer należy wybrać na podstawie wielu parametrów, m.in. oferowanych funkcji, pojemności konta, możliwości transferu, ceny.
- Opublikowanie strony polega na umieszczeniu jej na serwerze. Można do tego wykorzystać oferowane przez dostawcę oprogramowanie na serwerze lub użyć innych programów: klientów FTP bądź SCP.

## PYTANIA SPRAWDZAJĄCE

1. Na czym polega walidacja strony? Jakie mogą być konsekwencje zaniechania sprawdzenia strony przed jej opublikowaniem?
2. Jakie są najważniejsze kryteria wyboru usługi hostingowej?
3. W jaki sposób publikuje się stronę na serwerze? O czym trzeba pamiętać?

## PYTANIA QUIZOWE

1. Na czym polega sprawdzanie poprawności strony WWW?
   **a.** Na weryfikacji treści strony głównej.
   **b.** Na sprawdzaniu poprawności doboru kolorów na stronie.
   **c.** Na sprawdzaniu, czy strona jest responsywna.
   **d.** Na weryfikacji poprawności składniowej dokumentu HTML i arkusza CSS.
2. Z jakiego protokołu transferu plików korzysta się podczas przesyłania plików na serwer?
   **a.** FTP.
   **b.** HTTP.
   **c.** HTTPS.
   **d.** n.
3. Kto zajmuje się udostępnianiem stron internetowych w sieci?
   **a.** Graficy komputerowi.
   **b.** Webmasterzy.
   **c.** Firmy hostingowe.
   **d.** Administratorzy witryn.

# Indeks


## A

adres komórki 27–31, 38
  bezwzględny 31, 33, 38, 41
  mieszany 31, 38, 42
  względny 30–31, 38, 41
akapit 109, 110, 112, 114
aktualizacja 13, 19
alfabet Morse'a 198
algorytm
  C++ 241
  Python 183
algorytm naiwny wyszukiwania wzorca
  C++ 243
  Python 185–186
anagram
  C++ 250
  Python 196
anonimizacja 8
arkusz kalkulacyjny 26, 38
  graficzna prezentacja danych 33–35, 38
  historia zmian 65
  jako narzędzie symulacji 26, 49–50
  jako platforma zakupowa 56
  tabelaryczna prezentacja danych 38
  tryby wyświetlania formuł i wyników 28
  współdzielenie arkusza 61
  zastosowania arkusza 24–25
ASCII *patrz* kod ASCII
ataki *brute force* 12

## B

bezpieczeństwo komputerowe 11, 21
bękart 112
binarny sposób zapisu informacji 106
bitmapa 67
blog ekspercki 265–266, 273
blok
  C++ 213, 223
  Python 162, 164

## C

C++ 210–216, 218, 222
  ideone.com 212–214
  podstawowe konstrukcje 226
  typy danych 216, 233, 235, 249, 255
chmura 57
Code::Blocks 211–212
CSS 278, 292, 304–311, 313
CSV 144
cyberprzestępcy 11, 12
czcionka (edytor tekstu) 109
czcionka (HTML) 306–310
czekoladka 199

## D

dane 106
dane (arkusz kalkulacyjny) 38
  przedstawianie danych w formie graficznej 33–35, 38
dane (C++) 216, 233, 235, 249, 255
dane osobowe 8, 20
  wrażliwe 8
  zwykłe 8
data (arkusz kalkulacyjny) 28–29, 38
deklaracja zmiennych (C++) 216, 235
deszyfrowanie 199
  C++ 246, 254–255, 259
  Python 190, 200–201, 203–205
diagram *patrz* wykres
dokument o złożonej strukturze 132
  przygotowanie do druku *patrz* publikacja poligraficzna
dzielenie wyrazów 128, 130

## E

edytor grafiki 66–68
  GIMP 68–70
edytor równań (edytor tekstu) 128–130
edytor tekstu 108, 117
efekt czerwonych oczu 94
e-learning 265, 275–276, 279
  grywalizacja 262–263
ewaluacja 268

## F

filtr (edytor grafiki) 69, 74
formatowanie warunkowe 58–59, 61
formaty liczb 27, 38
formaty plików grafiki bitmapowej 79–80
  BMP 79
  GIF 80–83
  JPG 80
  PNG 80, 81
formuły (arkusz kalkulacyjny) 28, 38, 50
  kopiowanie formuł 30, 38
  warunkowe 40
formuły (C++) 240
forum dyskusyjne 265–267
FTP 325–326
funkcja
  C++ 219–220, 222–223, 226
  Python 161, 164, 167
funkcja (arkusz kalkulacyjny) 28–30, 38, 46, 61
  **DZIŚ** 29, 31
  **JEŻELI** 40–46, 52, 60
  kategorie funkcji 28
  kreator funkcji 30
  **LICZ.JEŻELI** 40, 43–44, 46
  logiczna 40
  matematyczna 28, 38
  **MAX** 29, 37
  **MIN** 29, 37, 52
  **MOD** 42
  **PI** 28, 32
  statystyczna 29, 38
  **SUMA** 28–29, 33, 45
  **SUMA.ILOCZYNÓW** 59–61
  **SUMA.JEŻELI** 40, 44–46
  **ŚREDNIA** 29, 37
  **TERAZ** 29
  **WYSZUKAJ.PIONOWO** 40, 47
  zagnieżdżona 41
funkcja **chr** (Python) 201–202
funkcja **main** (C++) 213, 222
funkcja **ord** (Python) 201–202
funkcja **range** (Python) *patrz* iterator **range**

## G

generator haseł 15
graficzna prezentacja danych 33–35, 38
grafika
  bitmapowa 67
  obiektowa 67
  rastrowa 66–68, 74
  wektorowa 67
grafika responsywna 309
grywalizacja 262–263

## H

hasło 10, 14–17, 21
  generator haseł 15
  menedżer haseł 14
  siła hasła 14, 16–17
  tester haseł 16–17
historia zmian (arkusz kalkulacyjny) 56
hosting 324–326
HTML 278, 289, 292, 304, 313, 315

## I

IDE
  C++ 210
  Python 155
ideone.com
  C++ 212, 214
  Python 156
IDLE 155–158, 212
indeks
  C++ 237, 239, 243, 245, 251
  Python 178, 180, 186, 188–189, 196
indeksowanie *patrz* indeks
indentacja (Python) 162, 164
infekcja 11
infografika (arkusz kalkulacyjny) 25
informacja (e-learning) 268, 273
  ocena informacji 268–269
  pozyskiwanie informacji 268, 273
  źródła informacji 265–269
instrukcja **cout** (C++) 213–214, 216, 235
instrukcja iteracji *patrz* pętla **for**
instrukcja **print** (Python) 157, 164, 176
instrukcja przypisania
  C++ 216
  Python 161, 167
instrukcja **return**
  C++ 220, 223
  Python 161, 164
instrukcja warunkowa (arkusz kalkulacyjny) 40–46

instrukcja warunkowa
C++ 208, 225, 225–228, 232–233
Python 152, 166–169, 174
interfejs użytkownika (arkusz kalkulacyjny) 52–54
lista rozwijana 56–57, 61
pasek przewijania 50, 52–53
*internet security* 13
interpreter (Python) 155, 164
iterator **range** (Python) 167, 171, 174

## J

JavaScript 283, 292, 315–321

## K

kaskadowe arkusze stylów *patrz* CSS
klucz
C++ 254–259
Python 200–201, 203–205
kod ASCII 107
C++ 255, 259
Python 201–202, 205
komentarz
C++ 213
HTML 293
Python 189
kompetencje medialne 264, 273
kompilator (C++) 210, 212
kompozycja zdjęcia 89, 98
zasada trójpodziału kadru 89, 91
kompresja 79–80
konkatenacja
C++ 236, 243
Python 177, 186
korespondencja seryjna 142–150
korygowanie zniekształceń 87
kryptografia
C++ 246, 259
Python 190, 205
kryptografia asymetryczna
C++ 259
Python 204–205
kształcenie zdalne *patrz* e-learning
kurs e-learningowy *patrz* e-learning
kurs online *patrz* e-learning

## L

layout 119–120
lista (Python) 180
lista rozwijana (arkusz kalkulacyjny) 57–58, 61
logowanie 17

## Ł

łamanie (edytor tekstu) 119, 125–128, 130
łamanie szyfru
C++ 258
Python 203

## M

*malware* 11
menedżer haseł 14
metatagi 293–294, 302
metoda (JavaScript) 316
metoda (Python) 179, 186
miękki Enter 128
MOOC 266, 280

## N

nagłówek (edytor tekstu) 135
nagłówek (HTML) 295–298, 301
napis (C++) 235–240, 242–253
analizowanie napisu 237
operacje na napisach 235–236, 243
przetwarzanie napisów 245
wyznaczanie długości napisu 236
napis (Python) 176–183, 185–197
analizowanie napisu 178, 186
operacje na napisach 176–177, 186
przetwarzanie napisów 188
wyznaczanie długości napisu 178, 186
nazwa komórki 50–51, 54
nazwa zakresu komórek 50–51, 54
netykieta 272–273

## O

obramowanie (edytor tekstu) 112, 114
ochrona
systemu i danych 11, 13
danych osobowych 8, 9
odnośniki wewnętrzne (HTML) 296

odszyfrowywanie wiadomości *patrz* deszyfrowanie
okładka
  książki tradycyjnej 101–104
  e-booka 102–104
okno Shell 155
operatory arytmetyczne
  C++ 215, 223
  Python 158, 164
operatory porównania
  C++ 218, 223
  Python 160, 164
optymalizowanie strony WWW 302–303
orientacja strony (edytor tekstu) 112–113, 117, 121

## P

palindrom
  C++ 248–251
  Python 192–196
parametry (C++) 219
parkan
  C++ 247–248, 251
  Python 191–192, 196
pasek przewijania (arkusz kalkulacyjny) 50, 52–53
pędzel 73–74, 95
pętla **for**
  C++ 226, 230, 233, 237, 243
  Python 167, 171, 174, 178, 186
pętla **while**
  C++ 208
  Python 152
phishing 12, 20
piksel 67, 74
pismo
  bezszeryfowe 109
  linie pisma 109
  szeryfowe 109
platforma e-learningowa 265, 273
plik wykonywalny (C++) 210
polecenie **cout** *patrz* instrukcja **cout**
polecenie **print** *patrz* instrukcja **print**
portal społecznościowy 265–267, 272–273
pozycjonowanie strony WWW 302–303
prawo autorskie 79, 136, 295
program antywirusowy 13, 21
programowanie
  C++ 243
  Python 186
pseudonimizacja 8
publikacja cyfrowa 103
  okładka 103, 104
  przygotowanie 138–140
publikacja poligraficzna 101
  grzbiet 101–102, 104
  okładka 101–104
  przygotowanie do druku 119, 132–138, 141
  skrzydełka 102
  spis treści 137, 141
  strona redakcyjna 136
  strona tytułowa 136
  wkład 101
publikowanie witryny WWW 322
Python 154, 164
  ideone.com 129, 133–134, 136
  IDLE 156
  podstawowe konstrukcje 167
  standardy kodowania 164

## R

rekurencja (JavaScript) 321
renderowanie 69, 83
retusz fotografii 85–87, 94, 98
  dodawanie efektów 97
  dodawanie elementów 95
  korygowanie zniekształceń 87–88
  usuwanie elementów 95
  zmiana kolorów 93–94
RODO 9
  pakiet praw konsumenta 9
ROT13 199

## S

samokształcenie 279
scalanie dokumentu (edytor tekstu) 148
sekcje (edytor tekstu) 133
sekwencja poleceń
  C++ 208, 225, 233
  Python 152, 166, 174
selekcjonowanie informacji (e-learning) 268, 273

serwis WWW *patrz* witryna WWW
  ekspercki 266, 273
sierota 112
skaner antywirusowy 13
skład (edytor tekstu) 119, 125, 130
składnia (Python) 156, 164
skoroszyt 26
skracanie linków 57
skrypt (Python) 156, 164
słowo kluczowe (Python) 168, 174
spad 102
sprawdzanie poprawności pisowni 115–117
  słownik standardowy i niestandardowy 117
stopka (edytor tekstu) 135
stopka (HTML) 295, 297, 301, 307
strona WWW *patrz* witryna WWW
struktura i wygląd dokumentu HTML 287, 292–302, 304–313
style (edytor tekstu) 125–126, 130
symulacja (arkusz kalkulacyjny) 26, 54
szablon (edytor tekstu) 119, 121–122, 146, 149
szablon (HTML) 286, 292
szerokość kolumn 121
szewc 112
szkolenie online *patrz* e-learning
szyfr asymetryczny *patrz* kryptografia asymetryczna
szyfr Cezara 199
  C++ 254–255, 258–259
  Python 200–201, 203, 205
szyfr podstawieniowy
  C++ 254
  Python 200
szyfr przestawieniowy
  C++ 246, 251
  Python 190, 196
szyfrogram 19, 199
  C++ 246, 251, 255, 259
  Python 190, 196, 201, 205
szyfrowanie 199
  C++ 251, 254–259
  Python 195–196, 200–205

## Ś

ścieżka aktywności (e-learning) 275
śledzenie zmian (arkusz kalkulacyjny) 56, 61

## T

tabulator 110–111, 117
tekst jawny 199
  C++ 246, 251, 255, 259
  Python 190, 196, 201, 204–205
tester haseł 16
testowanie
  C++ 225
  HTML 325
  Python 166
transfer 324, 326
trollowanie 272
twarda spacja 128
typografia 110

## U

uwierzytelnianie dwuskładnikowe 17–18, 21

## W

walidacja arkusza CSS 323, 325
walidacja kodu HTML 322–323, 325
warstwy 76–77, 83
  okno warstw 77
  warstwa tła 76, 83
  zasady pracy z warstwami 77
wartości logiczne
  C++ 182
  Python 160–161, 164, 168
wcięcia (Python) 162, 164
wdowa 112
*web usability* 285
witryna WWW 282–289, 302–303
  adaptywna 285–286, 289
  elementy dynamiczne *patrz* JavaScript
  funkcjonalność 285
  optymalizacja 302–303
  pozycjonowanie 302–303
  projektowanie 284, 289
  publikowanie 322–326
  responsywna 285–286, 289
  struktura strony *patrz* struktura i wygląd dokumentu HTML
  struktura 282–283, 287, 302–303
  wygląd strony *patrz* CSS oraz struktura i wygląd dokumentu HTML
współdzielenie arkusza 61
wstawianie wzorów 128–130

wtyczka 70
wycinanie 83
wykres 33–38
    kolumnowy 33
    kołowy 34
    liniowy 34
    pierścieniowy34
    słupkowy 33
    w postaci infografiki 25
wynik (Python) 164
wzorzec
    C++ 213, 242, 243
    Python 185, 186

## X

XML 144

## Z

zapora sieciowa 13
zasada trójpodziału kadru 89, 91
zintegrowane środowisko deweloperskie
    *patrz* IDE
złośliwe oprogramowanie *patrz* malware
zmienna sterująca pętlą
    C++ 230, 233
    Python 171, 174
zmienna
    C++ 215, 216, 223
    Python 161, 164
znaczniki HTML 289
znak zachęty (Python) 155

## Ź

źródła informacji 265–267

## Ż

żywa pagina 135

**Źródła ilustracji i fotografii**

**Okładka:** (kolorowe diagramy i wykresy) MicroOne/Shutterstock.com

**Jak korzystać z podręcznika:** (książki) Mary Long/Shutterstock.com

**Strony działowe:** s. 7 (bezpieczeństwo online) Mikko Lemola/Shutterstock.com; s. 23 (arkusz kalkulacyjny) lucadp/Shutterstock.com; s. 63 (słoneczniki) Sanchai Suksantidilog/Shutterstock.com; s. 105 (litery, cyfry, linie – abstrakcja) Zoonar GmbH/Alamy Stock Photo/BE&W; s. 151 (klawisz Python) Happy Stock Photo/Shutterstock.com; s. 207 (C++) StockEU/Shutterstock.com; s. 261 (laptop i schemat edukacji) JuliRose/Shutterstock.com; s. 281 (strona internetowa wyświetlana na laptopie, tablecie i smartfonie) REDPIXEL.PL/Shutterstock.com

**Infografiki:** s. 8 (kartka w linie) Forgem/Shutterstock.com; s. 9 (ikony z kartkami) Dstarky/Shutterstock.com, (ikony kosza i teczki) davooda/Shutterstock.com, (kolorowe ikony) gst/Shutterstock.com, (chłopak) BigLike Images/Shutterstock.com; s. 24 (biegnąca dziewczyna) Syda Productions/Shutterstock.com, (pieniądze) Inawa/Shutterstock.com; s. 25 (skład chemiczny człowieka) theromb/Shutterstock.com, (ręka z telefonem) Joe Besure/Shutterstock.com; s. 64 (polna droga, bociany) Mirosław Wyczółkowski/WSiP; s. 65 (zdjęcie niedoświetlone, zdjęcie prześwietlone) Mirosław Wyczółkowski/WSiP, (zrzut ze strony https://github.com/jantic/DeOldify); s. 100 (dziewczyna robiąca selfie) PanicAttack/Shutterstock.com, (słońce na niebie) sumroeng chinnapan/Shutterstock.com, (miasto) Eugenio Marongiu/Shutterstock.com, (ustalanie kadru) Shchus/Shutterstock.com, (ikony) Jarek Skawiński/Niokoba, s. 106 (neon zero, neon jeden) vlad_mlshk/Shutterstock.com; s. 107 (litery ABCD) vasabii/Shutterstock.com; s. 152 (samochody) Yuri Schmidt/Shutterstock.com, (jezdnie) Yuliia Konakhovska/Shutterstock.com, (policjant) artbesouro/Shutterstock.com; s. 153 (samochody) Yuri Schmidt/Shutterstock.com, (jezdnia) Yuliia Konakhovska/Shutterstock.com, (sygnalizacja świetlna) KeyVisualStock/Shutterstock.com, (drzewa) Anna Frajtova/Shutterstock.com, (ikona szkoły) Nadiinko/Shutterstock.com, (ikona sklepu) Leone_V/Shutterstock.com, (znak stopu) AStudio-1/Shutterstock.com, (kręta droga, pinezki) zmicier kavabata/Shutterstock.com; s. 184 (jajko) Pineapple studio/Shutterstock.com, (garnek) Gts/Shutterstock.com, (dzbanek z wodą) AtlasStudio/Shutterstock.com, (solniczka) Chones/Shutterstock.com, (błękitny minutnik) NY-P/Shutterstock.com, (kuchenka) Trovoboworod/Shutterstock.com, (jajowar) csivasz/Shutterstock.com, (jajka w wodzie) Teim/Shutterstock.com, (czerwony minutnik) Tatyana Pronina/Shutterstock.com, (tablet) GreenLandStudio/Shutterstock.com, (jajko na miękko) Pixel-Shot/Shutterstock.com; s. 193 (kobieta z laptopem) ESB Professional/Shutterstock.com; s. 198 (chorągiewki) Good Luck Photo/Shutterstock.com, (latarka) Yeti studio/Shutterstock.com, (telegraf) DutchScenery/Shutterstock.com; s. 199 (czekolada) Picsfive/Shutterstock.com; s. 208 (samochody) Yuri Schmidt/Shutterstock.com, (jezdnie) Yuliia Konakhovska/Shutterstock.com, (policjant) artbesouro/Shutterstock.com; s. 209 (samochody) Yuri Schmidt/Shutterstock.com, (jezdnia) Yuliia Konakhovska/Shutterstock.com, (sygnalizacja świetlna) KeyVisualStock/Shutterstock.com, (drzewa) Anna Frajtova/Shutterstock.com, (ikona szkoły) Nadiinko/Shutterstock.com, (ikona sklepu) Leone_V/Shutterstock.com, (znak stopu) AStudio-1/Shutterstock.com, (kręta droga, pinezki) zmicier kavabata/Shutterstock.com; s. 241 (jajko) Pineapple studio/Shutterstock.com, (garnek) Gts/Shutterstock.com, (dzbanek z wodą) AtlasStudio/Shutterstock.com, (solniczka) Chones/Shutterstock.com, (błękitny minutnik) NY-P/Shutterstock.com, (kuchenka) Trovoboworod/Shutterstock.com, (jajowar) csivasz/Shutterstock.com, (jajka w wodzie) Teim/Shutterstock.com, (czerwony minutnik) Tatyana Pronina/Shutterstock.com, (tablet) GreenLandStudio/Shutterstock.com, (jajko na miękko) Pixel-Shot/Shutterstock.com; s. 262 i 263 (ikony grywalizacji) Lemberg Vector studio/Shutterstock.com; s. 282 (dzieci na trawie) Pressmaster/Shutterstock.com; s. 283 (chłopiec) Iconic Bestiary/Shutterstock.com; s. 302 i 303 (robot z laptopem) kirill_makarov/Shutterstock.com

**Winietki:** s. 10 (bezpieczeństwo) sdecoret/Shutterstock.com; s. 26 (wykresy) tadamichi/Shutterstock.com; s. 40 (znaki zapytania) StepanPopov/Shutterstock.com; s. 49 (laptop) Africa Studio/Shutterstock.com; s. 56 (chmura) ra2studio/Shutterstock.com; s. 66 (górski szlak) Ruslan Suseynov/Shutterstock.com; s. 76 (warstwy agatu) Dafinchi/Shutterstock.com; s. 85 (kompozycja kwiatowa) Yulia Aksa/Shutterstock.com; s. 101 (e-book) iryna1/Shutterstock.com; s. 108 (ikony) merrilu/Shutterstock.com; s. 119 (tonery) pticelov/Shutterstock.com; s. 132 (dokumenty) NuPenDekDee/Shutterstock.com; s. 142 (kolorowe koperty) paulista/Shutterstock.com; s. 154 (logo języka Python) Logo Python jest znakiem towarowym firmy

Python Software Foundation, (tło pod logo – fragment kodu) BEST-BACKGROUNDS/Shutterstock.com; s. 166 (algorytm – abstrakcja) Visual Generation/Shutterstock.com; s. 176 (błękitne litery) bekulnis/Shutterstock.com; s. 188 (kostki z literami) cloud9works/Shutterstock.com; s. 200 (kod) BEST-BACKGROUNDS/Shutterstock.com; s. 210 (puzzle z C++) MoneyRender/Shutterstock.com; s. 225 (algorytm – abstrakcja) Aleksandr_a/Shutterstock.com; s. 235 (kolorowe litery) Alex Oakenman/Shutterstock.com; s. 245 (koraliki z literami) beast01/Shutterstock.com; s. 254 (szyfrogram) Sergei Drozd/Shutterstock.com; s. 264 (kod zerojedynkowy na autostradzie) metamorworks/Shutterstock.com; s. 275 (laptop-książka) Black Jack/Shutterstock.com; s. 284 (napis Web Design na ekranie) Rawpixel.com/Shutterstock.com; s. 292 (fragment kodu i neonowe figury) whiteMocca/Shutterstock.com; s. 304 (CSS na ekranie) one photo/Shutterstock.com; s. 315 (logo JavaScript) gdainti/Shutterstock.com, (rozmazany kod) Pattithi09/Shutterstock.com; s. 322 (walidacja) NicoElNino/Shutterstock.com;

**Zadania na start:** s. 10 (laptop) Chrisworo/Shutterstock.com; s. 40 (tabliczki) SilverCircle/Shutterstock.com; s. 49 (szybowiec) Weenee/Shutterstock.com, Lili White/Shutterstock.com; s. 56 (plansza do gry) Malachy666/Shutterstock.com; s. 76 (mapy natarcia) Adrian Bergiel; s. 85 (portret dziewczyny) Masson/Shutterstock.com; s. 101 (książki) Mary Long/Shutterstock.com s. 108 (kartka) Garsya/Shutterstock.com; s. 119 (monitor, broszura i ulotki) jukaspring/Shutterstock.com; s. 132 (pliki z brakującymi rozszerzeniami) Maxim Cherednichenko/Shutterstock.com; s. 142 (tenisówki) K.Decor/Shutterstock.com; s. 154 (chmura tagów – Python) Gordan/Shutterstock.com; s. 176 (DNA) ShadeDesign/Shutterstock.com; s. 188 (kryptex) Victor Moussa/Shutterstock.com; s. 200 (dziurka od klucza) sirius1/Shutterstock.com; s. 210 (chmura tagów – C++) Gordan/Shutterstock.com; s. 235 (DNA) Titov Nikolai/Shutterstock.com; s. 245 (kłódka) Volodymyr Krasyuk/Shutterstock.com, (mechanizm szyfrowy) Harper 3D/Shutterstock.com; s. 254 (dziurka od klucza) sirius1/Shutterstock.com; s. 264 (ikony serwisów społecznościowych) Artseen/Shutterstock.com; s. 284 (wyświetlanie kafelków na różnych ekranach) MPFphotography/Shutterstock.com; s. 292 (różne rodzaje kodu) monkographic/Shutterstock.com; s. 315 (stoper i lista) Telnov Oleksii/Shutterstock.com; s. 322 (symboliczna ujęcie budowy strony) liravega/Shutterstock.com

**Tekst główny:** s. 17 (klucz zabezpieczeń) Alamy; s. 22 praca przy komputerze (kadr z filmu); s. 67 (górski szlak, widok z górą i wrzosami) Ruslan Suseynov/Shutterstock.com; s. 73 (droga w górach) Ruslan Suseynov/Shutterstock.com; s. 79 (flaga Polski) PhotoStockImage/Shutterstock.com; s. 81 i 82 (zdjęcie z inscenizacji historycznej) Katarzyna Olędzka; s. 83 (pomnik Piłsudskiego) Stanislaw Tokarski/Shutterstock.com; s. 84 (dziewczyna z flagą) A StockStudio/Shutterstock.com; s. 86 (słonie) Katarzyna Olędzka; s. 87 (dziewczyna) B-D-S Piotr Marcinski/Shutterstock.com, (smartfon) Zyabich/Shutterstock.com, (świątynia Afai) Roman Sigaev/Shutterstock.com; s. 88 i 89 (żaglówka) Repina Valeriya/Shutterstock.com; s. 90 (tropikalna plaża) Galyna Andrushko/Shutterstock.com, (zachód słońca z palmami) Evgeniyqw/Shutterstock.com, (kobieta na plaży) OlegRi/Shutterstock.com; s. 91 (lew) Katarzyna Olędzka, (gepardy) Jen Watson/Shutterstock.com, (kitesurfer) Wallenrock/Shutterstock.com; s. 92 (kitesurfer) Wallenrock/Shutterstock.com; s. 93 (kitesurfer) Wallenrock/Shutterstock.com, (zachodzące słońce) Katarzyna Olędzka; s. 94 (efekt czerwonych oczu) Science Photo Library/East News; s. 95 (ptaki – montaż) mbolina/Shutterstock.com, (samolot – montaż) hxdyl/Shutterstock.com; s. 96 i 97 (kobieta na plaży) Kinga/Shutterstock.com; s. 98 (papugi) Only background/Shutterstock.com; s. 99 (świątynia Posejdona) Songquan Deng/Shutterstock.com, (łódź) Aerial-motion/Shutterstock.com, (Fira na Santorini) Zebra-Studio/Shutterstock.com, (przylądek Drastis) proslgn/Shutterstock.com, (uliczka na Kos) Nejdet Duzen/Shutterstock.com, (plaża Kastani na Santorini) leoks/Shutterstock.com; s. 102 (okładka *Księcia* – z lewej) © Wydawnictwo MG, (okładka *Księcia* – w środku) © Wydawnictwo MT Biznes, (okładka *Księcia* – z prawej) © Wydawnictwo MASTERLAB; s. 120 (świetlne napisy matematyczne) whiteMocca/Shutterstock.com, (symbole matematyki wewnątrz ołówka) jiris/Shutterstock.com, (pi, stożek, cyrkiel, ekierka) Marish/Shutterstock.com; s. 144 (ikona XML) vector illustration/Shutterstock.com; s. 149 (okładki książek A. Christie) Wydawnictwo Dolnośląskie, Publicat S.A.; s. 158, 161 i 171 (koła zębate) Mr. Rashad/Shutterstock.com; s. 201 (klucze) VectorShow/Shutterstock.com; s. 204 (dziewczyna z laptopem) Rido/Shutterstock.com, (klucze) VectorShow/Shutterstock.com, (koperta) rangizzz/Shutterstock.com, (chłopak z laptopem) Ollyy/Shutterstock.com; s. 215 i 216 (koła zębate) Mr. Rashad/Shutterstock.com; s. 224 (zegar) freedom100m/Shutterstock.com; s. 230 (koła zębate) Mr. Rashad/Shutterstock.com; s. 250 (planety) Irina Devaeva/Shutterstock.com; s. 255 (klucze) VectorShow/Shutterstock.com; s. 271 (uśmiechnięta dziewczyna) Cookie Studio/Shutterstock.com,

(samochód) Gold Picture/Shutterstock.com, (zdjęcie z basenu) Velislava Koleva/Shutterstock.com, (kot) Ro_ksy/Shutterstock.com, (wzór karty płatniczej) cosmic_pony/Shutterstock.com; s. 287 (dzieci na trawie) Pressmaster/Shutterstock.com; s. 295 (postać na tle kół zębatych) Tithi Luadthong/Shutterstock.com; s. 296 (okładka *The Time Machine* – pierwsza od lewej) The Protected Art Archive/Alamy Stock Photo/BE&W, (okładka *The Time Machine* – druga od lewej) GeoPic/Alamy Stock Photo/BE&W, (okładka *Wehikułu czasu* – druga od prawej) Wydawnictwo Wimana, (okładka *Wehikułu czasu* – pierwsza od prawej) Wydawnictwo CM; s. 309 (dzieci na trawie) Pressmaster/Shutterstock.com

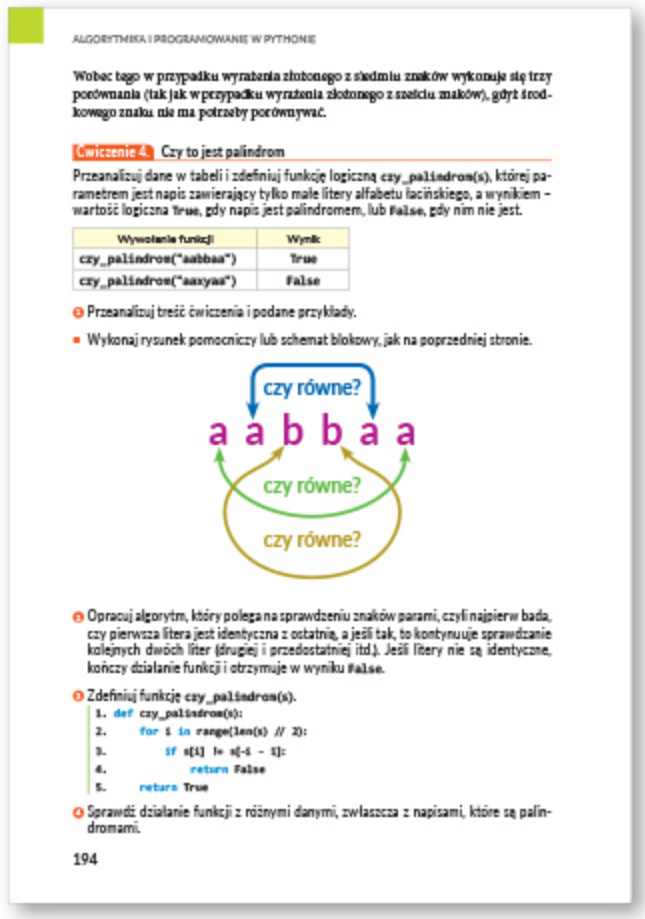
Wydawnictwa Szkolne i Pedagogiczne oświadczają, że podjęły starania, mające na celu dotarcie do właścicieli i dysponentów praw autorskich wszystkich zamieszczonych utworów. Wydawnictwa Szkolne i Pedagogiczne, przytaczając w celach dydaktycznych utwory lub fragmenty, postępują zgodnie z art. $27^1$ ustawy o prawie autorskim. Jednocześnie Wydawnictwa Szkolne i Pedagogiczne oświadczają, że są jedynym podmiotem właściwym do kontaktu autorów tych utworów lub innych podmiotów uprawnionych w wypadkach, w których twórcy przysługuje prawo do wynagrodzenia.

# INFORMATYKA ZAKRES PODSTAWOWY

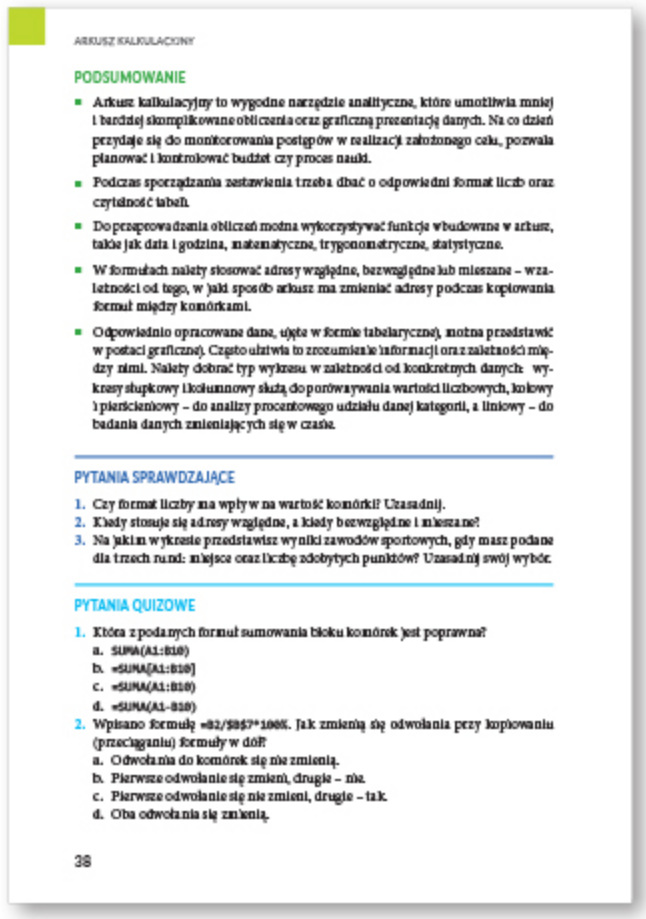
**NOWA EDYCJA** PODRĘCZNIK • LICEUM I TECHNIKUM




**Zadanie na start**
umożliwia samodzielną analizę problemu.




**Ćwiczenia krok po kroku**
pokazują zastosowanie omawianych zagadnień.




**Podsumowanie** i **pytania sprawdzające** oraz **testowe** systematyzują wiedzę.

Zeskanuj kod




# Aktywuj E-book

(cyfrowe odzwierciedlenie podręcznika) na EDURANGA


WSiP | wsip.pl
sklep.wsip.pl
infolinia: 801 220 555

EDURANGA

ORKE orke.pl