CENTRALNA KOMISJA EGZAMINACYJNA

Arkusz zawiera informacje prawnie chronione do momentu rozpoczęcia egzaminu.

WYPEŁNIA ZDAJĄCY

KOD | **PESEL**

Miejsce na naklejkę.
Sprawdź, czy kod na naklejce to **M-100**.
Jeżeli tak – przyklej naklejkę.
Jeżeli nie – zgłoś to nauczycielowi.

Egzamin maturalny ***Formuła 2023***

INFORMATYKA

Poziom rozszerzony

WYPEŁNIA ZDAJĄCY

WYBRANE:

..... (system operacyjny)

..... (program użytkowy)

..... (środowisko programistyczne)

Symbol arkusza
MINP-R0-**100**-2505

DATA: **14 maja 2025 r.**

GODZINA ROZPOCZĘCIA: **9:00**

CZAS TRWANIA: **210 minut**

LICZBA PUNKTÓW DO UZYSKANIA: **50**

Przed rozpoczęciem pracy z arkuszem egzaminacyjnym

1. Sprawdź, czy nauczyciel przekazał Ci **właściwy arkusz egzaminacyjny**, tj. arkusz we **właściwej formule**, z **właściwego przedmiotu** na **właściwym poziomie**.
2. Jeżeli przekazano Ci **niewłaściwy** arkusz – natychmiast zgłoś to nauczycielowi. Nie rozrywaj banderol.
3. Jeżeli przekazano Ci **właściwy** arkusz – rozerwij banderole po otrzymaniu takiego polecenia od nauczyciela. Zapoznaj się z instrukcją na stronie 2.

Układ graficzny
© CKE 2022

Kopia dokumentu CKE, oryginał: cke.gov.pl (pobrano 11.11.2026). Udostępnia: bitwizards.pl
Python, SQL i arkusz kalkulacyjny w przeglądarce, bez instalowania programów: bitwizards.pl

Instrukcja dla zdającego

1. Sprawdź, czy arkusz egzaminacyjny zawiera 18 stron (zadania 1–7) i czy dołączony jest do niego nośnik danych – podpisany DANE. Ewentualny brak zgłoś przewodniczącemu zespołu nadzorującego egzamin.
2. Na pierwszej stronie oraz na karcie odpowiedzi wpisz swój numer PESEL i przyklej naklejkę z kodem.
3. Wpisz zadeklarowane (wybrane) przez Ciebie na egzamin: system operacyjny, program użytkowy oraz środowisko programistyczne.
4. Symbol  zamieszczony w nagłówku zadania oznacza, że zadanie nie wymaga użycia komputera i odpowiedź do zadania należy zapisać tylko w miejscu na to przeznaczonym w arkuszu egzaminacyjnym.
5. Jeśli rozwiązaniem zadania lub jego części jest program komputerowy, to zapisz go w zadeklarowanym (wybranym) języku programowania i umieść w katalogu (folderze) oznaczonym Twoim numerem PESEL wszystkie utworzone przez siebie pliki w wersji źródłowej.
6. Jeżeli rozwiązaniem zadania lub jego części jest algorytm (który trzeba zapisać w arkuszu) i wybrałeś(-łaś) jego zapis w postaci języka programowania, to użyj języka programowania, który wybrałeś(-łaś) na egzamin (Java, C++ lub Python).
7. Jeśli rozwiązaniem zadania lub jego części jest baza danych utworzona z wykorzystaniem MySQL lub MariaDB, to umieść w katalogu (folderze) oznaczonym Twoim numerem PESEL treści zapytań w języku SQL oraz (przed zakończeniem egzaminu) wyeksportowaną całą bazę danych w formacie *.sql.
8. Pliki oddawane do oceny nazwij dokładnie tak, jak polecono w treści zadań, lub zapisz je pod nazwami (wraz z rozszerzeniem zgodnym z zadeklarowanym oprogramowaniem), jakie podajesz w arkuszu egzaminacyjnym. **Pliki o innych nazwach nie będą sprawdzane przez egzaminatora**.
   Pamiętaj, że zadania praktyczne niezawierające komputerowej realizacji rozwiązań zostaną ocenione na 0 punktów.
9. **Przed upływem czasu przeznaczonego na egzamin** zapisz w katalogu (folderze) oznaczonym Twoim numerem PESEL ostateczną wersję plików stanowiących rozwiązania zadań.
10. Pisz czytelnie. Używaj długopisu/pióra tylko z czarnym tuszem/atramentem.
11. Nie używaj korektora, a błędne zapisy wyraźnie przekreśl.
12. Nie wpisuj żadnych znaków w tabelkach przeznaczonych dla egzaminatora. Tabelki są umieszczone na marginesie przy każdym zadaniu.
13. Pamiętaj, że zapisy w brudnopisie nie będą oceniane.

Kopia dokumentu CKE, oryginał: cke.gov.pl (pobrano 11.11.2026). Udostępnia: bitwizards.pl
Każde zadanie programistyczne: najpierw plik przykładowy, potem pełne dane CKE - bitwizards.pl

Zadania egzaminacyjne są wydrukowane na następnych stronach.

Kopia dokumentu CKE, oryginał: cke.gov.pl (pobrano 11.11.2026). Udostępnia: bitwizards.pl
Python, SQL i arkusz kalkulacyjny w przeglądarce, bez instalowania programów: bitwizards.pl

Zadanie 1. Funkcja rekurencyjna

Dana jest rekurencyjna funkcja *przestaw*, której parametrem jest nieujemna liczba całkowita:

```
przestaw(n):
    r ← n mod 100
    a ← r div 10
    b ← r mod 10
    n ← n div 100
    jeżeli n > 0
        w ← a + 10 * b + 100 * przestaw(n)
    w przeciwnym razie
        jeżeli a > 0
            w ← a + 10 * b
        w przeciwnym razie
            w ← b
    wynikiem jest w
```

Uwaga:
Operator mod oznacza resztę z dzielenia, natomiast div – część całkowitą z dzielenia.

1.1.
0–1–2–3

Zadanie 1.1. (0–3)

Uzupełnij tabelę – wpisz w drugiej kolumnie wynik funkcji *przestaw*(*n*) dla podanych wartości argumentu *n* oraz wpisz w trzeciej kolumnie liczbę wywołań funkcji *przestaw* łącznie z pierwszym wywołaniem z parametrem *n*.

n	**Wynik działania funkcji *przestaw***	**Liczba wywołań funkcji *przestaw***
316498	134689	3
43657688		
154005710		
998877665544321		

Miejsce na obliczenia (brudnopis)

Kopia dokumentu CKE, oryginał: cke.gov.pl (pobrano 11.11.2026). Udostępnia: bitwizards.pl
Każde zadanie programistyczne: najpierw plik przykładowy, potem pełne dane CKE - bitwizards.pl

Zadanie 1.2. (0–2)

1.2.
0–1–2

Oceń prawdziwość podanych zdań. Zaznacz **P**, jeśli zdanie jest prawdziwe, albo **F** – jeśli jest fałszywe.

Niech *n* będzie liczbą *k*-cyfrową, gdzie $k > 0$. Liczba wywołań funkcji *przestaw* w zależności od *k* jest równa:

1.	$\frac{k}{2}$	**P**	**F**
2.	$(k+1) \operatorname{div} 2$ (gdzie div oznacza dzielenie całkowite)	**P**	**F**
3.	$\begin{cases} \frac{k}{2} & \text{gdy } k \text{ jest liczbą parzystą} \\ \frac{k+1}{2} & \text{gdy } k \text{ jest liczbą nieparzystą} \end{cases}$	**P**	**F**
4.	$\frac{k+1}{2}$	**P**	**F**

Miejsce na obliczenia (brudnopis)

Kopia dokumentu CKE, oryginał: cke.gov.pl (pobrano 11.11.2026). Udostępnia: bitwizards.pl
Rozwiązania tego arkusza krok po kroku, w przeglądarce: bitwizards.pl/probuj/

1.3.
0–1–2–3–4

Zadanie 1.3. (0–4)

W postaci pseudokodu lub w wybranym języku programowania napisz **nierekurencyjną** funkcję *przestaw2*, która dla danej nieujemnej liczby całkowitej *n* da taką samą wartość jak *przestaw*(*n*).

Uwaga: Twój algorytm może używać **wyłącznie zmiennych przechowujących liczby całkowite** oraz może operować **wyłącznie na liczbach całkowitych**. W zapisie możesz wykorzystać tylko operacje arytmetyczne: dodawanie, odejmowanie, mnożenie, dzielenie, dzielenie całkowite, resztę z dzielenia oraz porównywanie liczb, instrukcje sterujące, przypisania do zmiennych lub samodzielnie napisane funkcje, wykorzystujące wyżej wymienione operacje. **Zabronione** jest używanie funkcji wbudowanych oraz operatorów innych niż wymienione, **w tym – funkcji *przestaw***.

Specyfikacja:

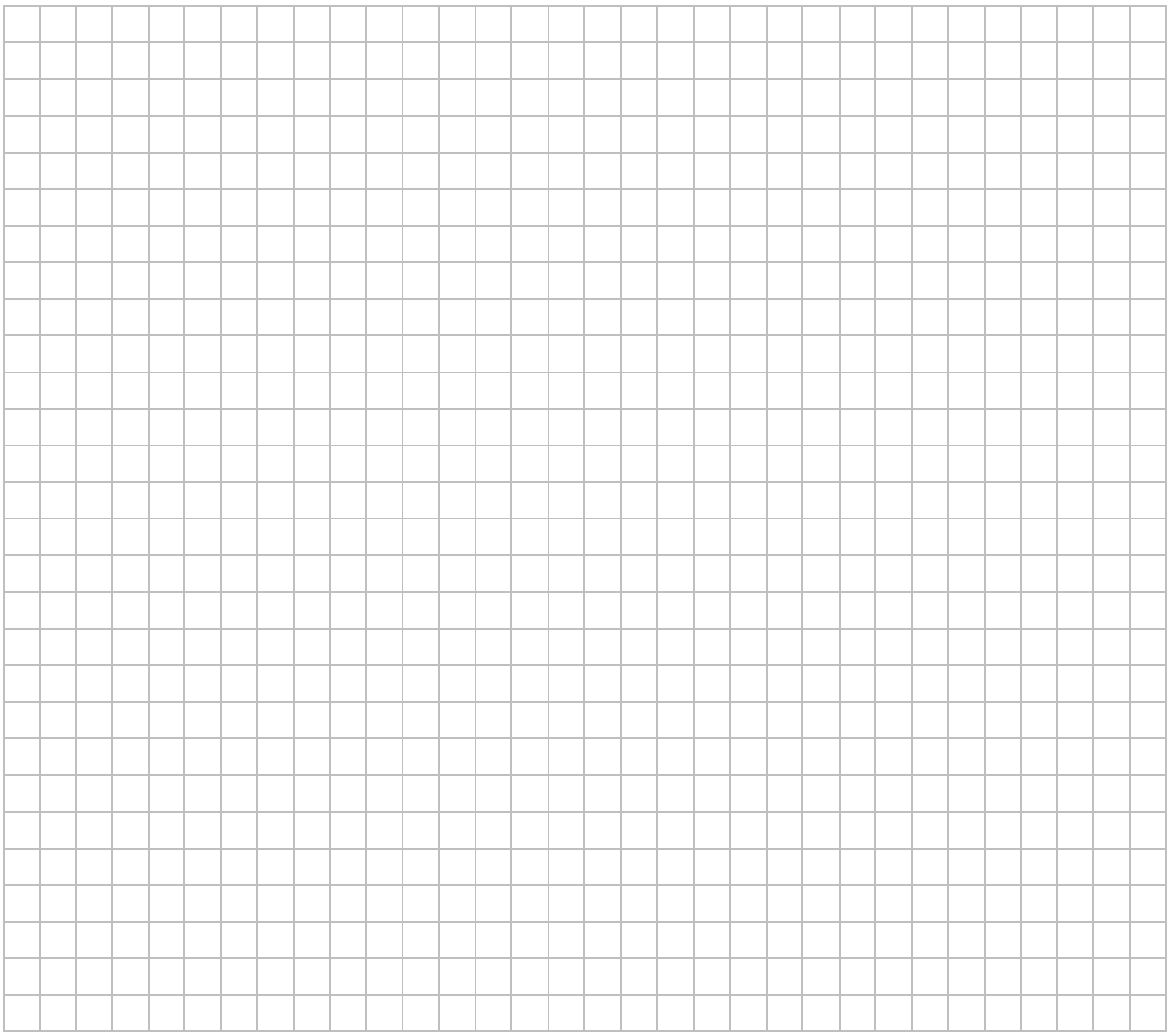
Dane

n – nieujemna liczba całkowita

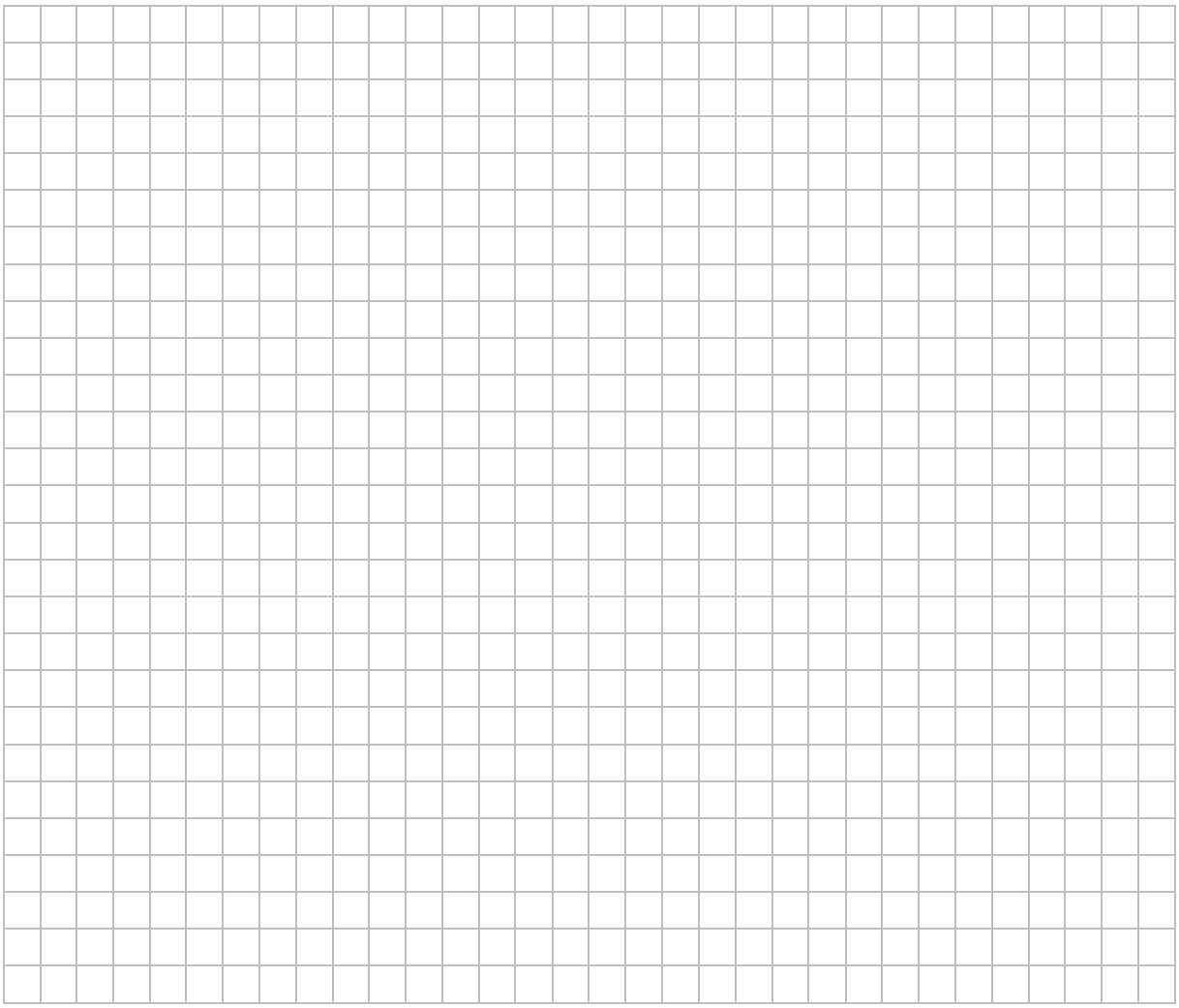
Wynik

w – nieujemna liczba całkowita, wynik działania taki sam jak po wykonaniu *przestaw*(*n*)

Miejsce na zapis algorytmu



Kopia dokumentu CKE, oryginał: cke.gov.pl (pobrano 11.11.2026). Udostępnia: bitwizards.pl
Pytania z egzaminów zawodowych z wyjaśnieniami odpowiedzi: zawodowe.edu.pl



Kopia dokumentu CKE, oryginał: cke.gov.pl (pobrano 11.11.2026). Udostępnia: bitwizards.pl
Pytania z egzaminów zawodowych z wyjaśnieniami odpowiedzi: zawodowe.edu.pl

Zadanie 2. Zapis symboliczny

W pliku `symbole.txt` zapisano 2000 napisów. Każdy z nich jest zapisany w osobnym wierszu i składa się z dokładnie 12 znaków spośród: `o`, `+`, `*`.

Napisz program (lub kilka programów) znajdujący(-ch) odpowiedzi do podanych zadań. Każdą odpowiedź zapisz w pliku `wyniki2.txt` i poprzedź ją numerem oznaczającym zadanie.

Do Twojej dyspozycji jest plik `symbole_przyklad.txt`, który zawiera 20 wierszy danych spełniających warunki zadania. Odpowiedzi dla pliku `symbole_przyklad.txt` są podane pod każdym zadaniem.

Pamiętaj, że Twój program musi ostatecznie zadziałać na pliku `symbole.txt` zawierającym 2000 napisów.

2.1. 0–1–2

Zadanie 2.1. (0–2)

Podaj wszystkie takie napisy z pliku `symbole.txt`, które są palindromami (czytane od przodu i od tyłu są takie same). Wypisz je po jednym w wierszu, w kolejności takiej jak w pliku `symbole.txt`.

Odpowiedź dla pliku `symbole_przyklad.txt` to
`oooo+**+oooo`
(w tym pliku jest jeden palindrom)

2.2. 0–1–2–3–4

Zadanie 2.2. (0–4)

W pliku `symbole.txt` szukamy „kwadratów” złożonych z dziewięciu sąsiadujących identycznych symboli:

```
+ + +        o o o        * * *
+ + +  lub   o o o  lub   * * *
+ + +        o o o        * * *
```

Podaj, ile takich kwadratów występuje w pliku `symbole.txt`. Jeżeli w pliku występuje jeden taki kwadrat, podaj numer wiersza i numer pozycji w wierszu (licząc od 1) jego **środkowego pola**. Jeżeli jest więcej takich kwadratów, podaj numer wiersza i numer pozycji w wierszu dla środkowego pola każdego z nich.

Przykład:
Poniżej podano 6 wierszy przykładowych danych (po 12 znaków w każdym wierszu):
1. `+**+o*o+++*o+`
2. `+++ooooo*o***`
3. `+o*oooo**+++`
4. `*+*oooooo+++`
5. `o**o+++o++++`
6. `oooo++**+*+o`

Kopia dokumentu CKE, oryginał: cke.gov.pl (pobrano 11.11.2026). Udostępnia: bitwizards.pl
Każde zadanie programistyczne: najpierw plik przykładowy, potem pełne dane CKE - bitwizards.pl

Mamy tutaj trzy kwadraty złożone z 9 identycznych symboli: pierwszy ma środek w wierszu 3 na pozycji 5, drugi – w wierszu 3 na pozycji 6, a trzeci – w wierszu 4 na pozycji 11.

Odpowiedź dla pliku `symbole_przyklad.txt` to
1 6 3
(jeden kwadrat, który ma środkowe pole w wierszu 6, na pozycji 3).

Informacja do zadań 2.3. i 2.4.

Każdy z napisów podanych w pliku `symbole.txt` będziemy traktować jako liczbę zapisaną w systemie trójkowym, w którym:
znak `o` odpowiada cyfrze 0
znak `+` odpowiada cyfrze 1
znak `*` odpowiada cyfrze 2.

Zadanie 2.3. (0–2)

2.3.
0–1–2

Podaj największą liczbę spośród liczb zapisanych w pliku `symbole.txt`. W odpowiedzi podaj tę liczbę w zapisie dziesiętnym oraz napis jej odpowiadający.

Odpowiedź dla pliku `symbole_przyklad.txt` to
519789 `***+o*ooo++o`

Zadanie 2.4. (0–3)

2.4.
0–1–2–3

Oblicz sumę wszystkich liczb z pliku `symbole.txt`. Podaj jej wartość w zapisie dziesiętnym oraz w zapisie trójkowym z użyciem symboli: `o`, `+`, `*`.

Odpowiedź dla pliku `symbole_przyklad.txt` to
4841542 `+oooo****+oo+o+`

Do oceny oddajesz:

- plik `wyniki2.txt` – zawierający odpowiedzi do zadań 2.1.–2.4.
  (odpowiedź do każdego zadania powinna być poprzedzona jego numerem)
- pliki zawierające kody źródłowe Twojego(-ich) programu(-ów) o nazwach
  (uwaga: brak tych plików jest równoznaczny z brakiem rozwiązania zadania):

  zadanie 2.1. ……

  zadanie 2.2. ……

  zadanie 2.3. ……

  zadanie 2.4. ……

Kopia dokumentu CKE, oryginał: cke.gov.pl (pobrano 11.11.2026). Udostępnia: bitwizards.pl
Rozwiązania tego arkusza krok po kroku, w przeglądarce: bitwizards.pl/probuj/

Zadanie 3. Dron

Tor lotu pewnego drona składa się z prostych odcinków. Lot rozpoczyna się w punkcie (0, 0), a kończy w punkcie (20000, 0). Dron poza startem i lądowaniem jest zawsze na wysokości większej od zera.

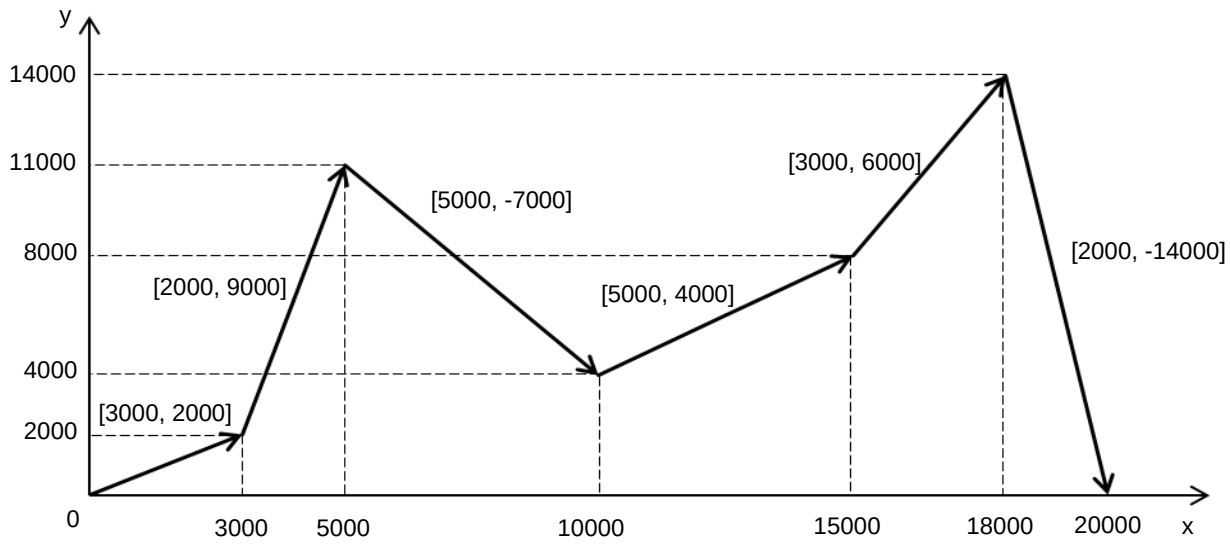
Plik `dron.txt` zawiera 100 wierszy, w których zapisano dane dotyczące ruchu drona. W każdym wierszu jest zapisana para liczb całkowitych rozdzielonych znakiem spacji. Pierwsza liczba oznacza przemieszczenie drona (odległość) w poziomie od ostatniej pozycji – jest to zawsze liczba dodatnia. Druga liczba oznacza przemieszczenie w pionie od ostatniej pozycji. Jeśli druga liczba jest dodatnia, to dron wykonał ruch w górę, jeśli ujemna – w dół, a jeśli równa 0 – nie zmieniał wysokości.

Przykład 1.

Dla przykładowych danych:

3000 2000
2000 9000
5000 -7000
5000 4000
3000 6000
2000 -14000

lot drona można zilustrować na wykresie:



gdzie:

x – odległość w poziomie od punktu startowego

y – wysokość (odległość w pionie od punktu startowego)

[*A*, *B*] – umieszczone na wykresie pary liczb oznaczające przemieszczenia drona odpowiednio w poziomie i w pionie.

Napisz program (lub kilka programów), który(-e) znajdzie(-dą) odpowiedzi dla podanych zadań. Każdą odpowiedź zapisz w pliku `wyniki3.txt` i poprzedź ją numerem oznaczającym zadanie.

Kopia dokumentu CKE, oryginał: cke.gov.pl (pobrano 11.11.2026). Udostępnia: bitwizards.pl
Każde zadanie programistyczne: najpierw plik przykładowy, potem pełne dane CKE - bitwizards.pl

Do Twojej dyspozycji jest plik `dron_przyklad.txt` zawierający 10 wierszy danych w opisanej postaci. Odpowiedzi dla pliku `dron_przyklad.txt` są podane pod każdym zadaniem. Pamiętaj, że Twój program musi ostatecznie zadziałać na pliku `dron.txt` zawierającym 100 wierszy danych.

Zadanie 3.1. (0–2)

3.1.
0–1–2

Dla każdego przesunięcia [*A*, *B*] zapisanego w pliku `dron.txt` oblicz największy wspólny dzielnik (NWD) wartości bezwzględnych liczb *A* i *B*. Podaj **liczbę** par [*A*, *B*], dla których największy wspólny dzielnik wartości bezwzględnych liczb *A* i *B* jest większy od 1.

Uwaga: przyjmujemy, że NWD(*A*, 0) = *A*.

Odpowiedź dla pliku `dron_przyklad.txt` to
6

Zadanie 3.2. (0–4)

3.2.
0–1–2–3–4

Rozważmy **wszystkie punkty**, w których dron znajdował się **po wykonaniu** kolejnych ruchów (przesunięć).

Dla danych z przykładu 1. będą to punkty: (3000, 2000), (5000, 11000), (10000, 4000), (15000, 8000), (18000, 14000) i (20000, 0).

a) Podaj, ile spośród wszystkich rozważanych punktów znajduje się wewnątrz kwadratu o wierzchołkach (0, 0), (0, 5000), (5000, 5000), (5000, 0). Nie liczymy punktów leżących na krawędziach kwadratu.

b) Spośród wszystkich rozważanych punktów znajdź i podaj trzy różne, takie, że jeden z nich jest środkiem odcinka o końcach w pozostałych dwóch. Jest tylko jedna taka trójka punktów.
Uwaga: punkty należące do szukanej trójki nie muszą być trzema kolejnymi punktami, do których przemieszczał się dron.

Odpowiedź dla pliku `dron_przyklad.txt` to
a) 2
b) (14000, 3014), (16000, 2010), (18000, 1006)

Do oceny oddajesz:

- plik `wyniki3.txt` – zawierający odpowiedzi do zadań 3.1.–3.2.
  (odpowiedź do każdego zadania powinna być poprzedzona jego numerem)
- pliki zawierające kody źródłowe Twojego(-ich) programu(-ów) o nazwach
  (uwaga: brak tych plików jest równoznaczny z brakiem rozwiązania zadania):

  zadanie 3.1. ……

  zadanie 3.2. ……

Kopia dokumentu CKE, oryginał: cke.gov.pl (pobrano 11.11.2026). Udostępnia: bitwizards.pl
Python, SQL i arkusz kalkulacyjny w przeglądarce, bez instalowania programów: bitwizards.pl

4.
0–1

Zadanie 4. (0–1)

Dokończ zdanie. Zaznacz właściwą odpowiedź spośród podanych.

Program typu *keylogger* służy do

A. szyfrowania informacji do postaci uniemożliwiającej jej odczytanie bez zdefiniowanego klucza.
B. przechowywania danych logowania, w tym haseł, w bezpiecznym miejscu na dysku użytkownika.
C. generowania kodu, który umożliwia użytkownikowi bankowości elektronicznej wykonanie operacji.
D. przechwytywania i gromadzenia informacji o naciśniętych klawiszach.

5.
0–1–2

Zadanie 5. (0–2)

Poniżej sposobem pisemnym dodano dwie liczby podane w zapisie binarnym. Uzupełnij brakujące cyfry tak, aby działanie było wykonane poprawnie.

$$\begin{array}{rcccccccccc} & 1 & 1 & \square & 0 & 1 & 0 & 1 & 1 & 0 & \square & 1 \\ + & & 1 & 1 & 0 & 0 & \square & 1 & 0 & 1 & 1 & 1 \\ \hline 1 & \square & 0 & 1 & 1 & 0 & 0 & 1 & \square & \square & 1 & 0 \end{array}$$

Miejsce na obliczenia (brudnopis)



Kopia dokumentu CKE, oryginał: cke.gov.pl (pobrano 11.11.2026). Udostępnia: bitwizards.pl
Rozwiązania tego arkusza krok po kroku, w przeglądarce: bitwizards.pl/probuj/

Zadanie 6. Martianeum

W 2033 roku na Marsie wylądowała automatyczna stacja wydobywcza wyposażona w transporter i w autonomiczny dron pobierający ładunki skał zawierających minerał niewystępujący na Ziemi – martianeum.

Stacja działa według następujących zasad:

- dron codziennie przywozi ładunek z pewnego obszaru Marsa
- stacja waży ładunek przywieziony przez drona i bada zawartość martianeum
- jeśli zawartość martianeum w przywiezionym ładunku wynosi co najmniej 1%, to stacja automatycznie wydobywa cały minerał z tego ładunku
- jeśli na koniec dnia (po wydobyciu martianeum) ilość minerału na stacji osiągnie co najmniej 100 kg, to transporter zabiera 100 kg na orbitę, skąd ładunek jest wysyłany na Ziemię, a transporter wraca do stacji (jeśli na stacji zgromadzone jest więcej niż 100 kg, to nadmiar pozostaje na stacji)
- początkowy stan magazynu na stacji – 0 kg martianeum.

W pliku tekstowym `martianeum.txt` w kolejnych wierszach zapisano dane z lat 2033–2038:

`data` – data przywozu ładunku w formacie rrrr-mm-dd
`nazwa_obszaru` – nazwa obszaru Marsa, z którego ładunek został pobrany
`masa [kg]` – masa ładunku drona w kilogramach
`zawartość [%]` – zawartość martianeum w próbce w % (nieujemna liczba z jednym miejscem po przecinku, np. 0,1 oznacza 0,1%)

Dane w pliku rozdzielono znakami tabulacji.

Przykład:

```
data          nazwa_obszaru      masa [kg]      zawartosc [%]
2033-03-03    Cebrenia           27,8           0,2
2033-03-04    Amenthes           11,8           1,7
2033-03-05    Noachis            21,0           6,0
2033-03-06    Coprates           26,3           11,4
2033-03-07    Ismenius Lacus     28,8           0,0
2033-03-08    Mare Boreum        29,2           0,0
```

Z wykorzystaniem danych zawartych w pliku `martianeum.txt` oraz dostępnych narzędzi informatycznych wykonaj podane zadania. Wyniki zapisz w pliku tekstowym `wyniki6.txt`. Odpowiedź do każdego zadania poprzedź numerem tego zadania.

Zadanie 6.1. (0–2)

Podaj łączną masę ładunków drona oraz łączną masę martianeum wydobytego przez stację.

6.1.
0–1–2

Kopia dokumentu CKE, oryginał: cke.gov.pl (pobrano 11.11.2026). Udostępnia: bitwizards.pl
Każde zadanie programistyczne: najpierw plik przykładowy, potem pełne dane CKE - bitwizards.pl

6.2.
0–1

Zadanie 6.2. (0–1)

Podaj nazwę obszaru, dla którego średnia masa przywiezionych ładunków jest najmniejsza.

6.3.
0–1–2

Zadanie 6.3. (0–2)

Czas pracy stacji dzielimy na kolejne 7-dniowe okresy. Pierwszy okres obejmuje dni od 03.03.2033 do 09.03.2033, drugi – od 10.03.2033 do 16.03.2033 itd.
Podaj największą łączną masę ładunków przywiezionych w ciągu kolejnych 7-dniowych okresów oraz podaj datę początku okresu, w którym przywieziono tę największą masę.

6.4.
0–1–2–3

Zadanie 6.4. (0–3)

Wykonaj zestawienie, w którym dla każdego obszaru podasz, ile razy dron przewoził ładunek z tego obszaru w poszczególnych latach.
Na podstawie wykonanego zestawienia utwórz wykres skumulowany kolumnowy. Pamiętaj o czytelnym opisie wykresu: na osi X umieść nazwy obszarów, dodaj opisy osi – „nazwy obszarów” dla osi X i „liczba przewozów ładunku” dla osi Y, tytuł oraz legendę zawierającą kolejne lata.

6.5.
0–1–2–3

Zadanie 6.5. (0–3)

Uwzględnij zasady działania stacji opisane na początku zadania i podaj:

- ile razy stacja wysyłała ładunek na orbitę
- datę pierwszego transportu ładunku na orbitę
- datę ostatniego transportu ładunku na orbitę.

Do oceny oddajesz:

- plik tekstowy `wyniki6.txt` zawierający odpowiedzi do zadań 6.1.–6.5. Odpowiedź do każdego zadania powinna być poprzedzona jego numerem.
- plik zawierający wykres do zadania 6.4. o nazwie:

  ……
- plik(pliki) zawierający(-e) komputerową realizację Twoich rozwiązań o nazwie(-ach): (uwaga: brak tych plików jest równoznaczny z brakiem rozwiązania zadania)

  .....

Kopia dokumentu CKE, oryginał: cke.gov.pl (pobrano 11.11.2026). Udostępnia: bitwizards.pl
Pytania z egzaminów zawodowych z wyjaśnieniami odpowiedzi: zawodowe.edu.pl

Zadanie 7. Poszukiwanie wody na Marsie

W trzech plikach tekstowych o nazwach `laziki.txt`, `obszary.txt`, `pomiary.txt` zapisano informacje zawierające dane o poszukiwaniu wody na Marsie w latach 2050–2080. Łaziki zasilane energią słoneczną poruszają się po różnych obszarach Marsa i wykonują pomiary georadarowe, na podstawie których szacują ilość wody i głębokość, na której się ona znajduje. Pierwszy wiersz każdego z plików jest wierszem nagłówkowym, a dane w wierszach rozdzielono znakami tabulacji.

Plik o nazwie `laziki.txt` zawiera informacje o różnych łazikach, które wykonywały pomiary. W każdym wierszu tego pliku znajdują się:

`nr_lazika` – co najwyżej trzycyfrowy, unikatowy numer łazika
`nazwa_lazika` – nazwa łazika (tekst do 50 znaków)
`rok_wyslania` – rok startu z Ziemi
`wsp_ladowania` – współrzędne lądowania na Marsie oddzielone znakiem przecinka i spacją

Przykład:

```
nr_lazika nazwa_lazika   rok_wyslania   wsp_ladowania
1         Mariner 3      2049           50.51N, 70.01E
2         Mariner 6      2050           11.90N, 119.49E
3         Mariner 7      2050           44.90S, 130.80W
```

Plik o nazwie `obszary.txt` zawiera informacje o obszarach na Marsie. W każdym wierszu tego pliku znajdują się:

`kod_obszaru` – pięcioznakowy, unikatowy kod obszaru
`nazwa_obszaru` – nazwa obszaru (tekst do 50 znaków)

Przykład:

```
kod_obszaru    nazwa_obszaru
MC-01          Mare Boreum
MC-02          Diacria
MC-03          Arcadia
```

Plik o nazwie `pomiary.txt` zawiera informacje o wynikach badań georadarowych wykonanych przez łaziki. W każdym wierszu tego pliku znajdują się:

`nr_lazika` – co najwyżej trzycyfrowy numer łazika
`data_pomiaru` – data wykonania pomiaru (w formacie rrrr-mm-dd)
`kod_obszaru` – pięcioznakowy kod obszaru, na którym został wykonany pomiar
`wspolrzedne` – współrzędne wykonania pomiaru, oddzielone znakiem przecinka i spacją
`glebokosc` – szacowana głębokość, na której znajduje się woda (w metrach)
`ilosc` – szacowana ilość wody (w m^3)

Kopia dokumentu CKE, oryginał: cke.gov.pl (pobrano 11.11.2026). Udostępnia: bitwizards.pl
Arkusze i klucze egzaminów zawodowych INF.02, INF.03 i INF.04: zawodowe.edu.pl

Przykład:

```
nr_      data_        kod_    wspolrzedne      glebokosc ilosc
lazika   pomiaru      obszaru
17       2061-06-03   MC-13   13.17N, 77.80E   344       5622
17       2056-06-02   MC-14   14.93N, 106.00E  43        2054
47       2075-10-18   MC-05   45.57N, 3.30E    9         23366
```

Z wykorzystaniem danych zawartych w podanych plikach oraz dostępnych narzędzi informatycznych podaj odpowiedzi do zadań 7.1–7.4. Odpowiedzi zapisz w pliku `wyniki7.txt`, a każdą z nich poprzedź numerem odpowiedniego zadania.

7.1. 0–1–2

Zadanie 7.1. (0–2)

Podaj nazwę obszaru, na którym znaleziono łącznie we wszystkich pomiarach najwięcej m^3 wody na głębokości do 100 metrów włącznie. Jest jeden taki obszar.

7.2. 0–1–2

Zadanie 7.2. (0–2)

Podaj nazwę łazika, który wykonywał pomiary w najdłuższym okresie, licząc od pierwszego (najwcześniejszego) do ostatniego (najpóźniejszego) pomiaru. Podaj datę pierwszego i ostatniego pomiaru wykonanego przez ten łazik.

7.3. 0–1–2

Zadanie 7.3. (0–2)

Podaj nazwy obszarów na Marsie, na których żaden z łazików nie wykonał ani jednego pomiaru w tym samym roku, w którym został wysłany z Ziemi.

7.4. 0–1–2

Zadanie 7.4. (0–2)

Podaj nazwy łazików, które wylądowały na półkuli południowej, ale wykonywały pomiary na obu półkulach: północnej (N) i południowej (S).

Do oceny oddajesz:

- plik tekstowy `wyniki7.txt` – zawierający odpowiedzi do poszczególnych zadań (odpowiedź do każdego zadania powinna być poprzedzona jego numerem)
- plik(i) zawierający(e) komputerową realizację Twoich obliczeń o nazwie(-ach) (uwaga: brak tych plików jest równoznaczny z brakiem rozwiązania zadania):

.....

Kopia dokumentu CKE, oryginał: cke.gov.pl (pobrano 11.11.2026). Udostępnia: bitwizards.pl
Kurs: Matura z informatyki - arkusze krok po kroku, bez logowania: bitwizards.pl

Zadanie 7.5. (0–2)

7.5.
0–1–2

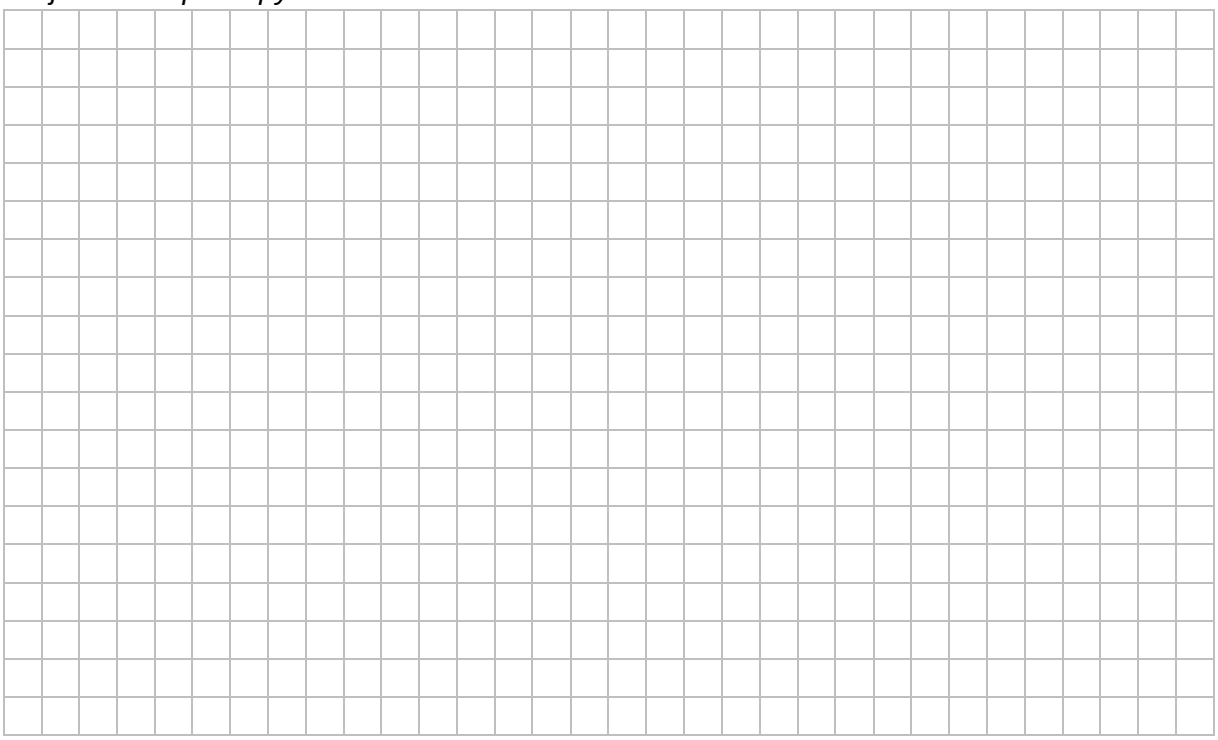
Do tabel utworzonych na podstawie opisanych wcześniej plików dołączamy kolejną – o nazwie *Producent*, w której zapisano informacje o producentach poszczególnych modeli łazików.

Tabela *Producent* zawiera następujące pola:
`kod_producenta` – unikatowy kod producenta
`nazwa` – nazwa producenta
`kraj` – kraj producenta

Do tabeli *Laziki* dodano pole `kod_producenta`.

Napisz w języku SQL zapytanie, w wyniku którego otrzymasz listę nazw producentów, których łaziki badały obszar Marsa o nazwie Arcadia w roku 2060. Nazwy producentów nie mogą się powtarzać.

Miejsce na zapis zapytania



Kopia dokumentu CKE, oryginał: cke.gov.pl (pobrano 11.11.2026). Udostępnia: bitwizards.pl
Pytania z egzaminów zawodowych z wyjaśnieniami odpowiedzi: zawodowe.edu.pl

BRUDNOPIS (*nie podlega ocenie*)

Kopia dokumentu CKE, oryginał: cke.gov.pl (pobrano 11.11.2026). Udostępnia: bitwizards.pl
Algorytmy z podstawy programowej opisane krok po kroku: bitwizards.pl/algorytmy/

Kopia dokumentu CKE, oryginał: cke.gov.pl (pobrano 11.11.2026). Udostępnia: bitwizards.pl
Każde zadanie programistyczne: najpierw plik przykładowy, potem pełne dane CKE - bitwizards.pl

INFORMATYKA

Poziom rozszerzony

Formuła 2023



INFORMATYKA

Poziom rozszerzony

Formuła 2023



INFORMATYKA

Poziom rozszerzony

Formuła 2023



Kopia dokumentu CKE, oryginał: cke.gov.pl (pobrano 11.11.2026). Udostępnia: bitwizards.pl
Python, SQL i arkusz kalkulacyjny w przeglądarce, bez instalowania programów: bitwizards.pl