

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**-2605

DATA: **14 maja 2026 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.





Instrukcja dla zdającego

1. Sprawdź, czy arkusz egzaminacyjny zawiera 21 stron (zadania 1–8) i czy dołączony jest do niego nośnik danych – podpisany DANE. Ewentualny brak zgłoś przewodniczącemu zespołu nadzorującego egzamin.
2. Na pierwszej stronie arkusza 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 wybierasz jego zapis w postaci języka programowania, to użyj tego języka programowania, który został przez Ciebie wybrany 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.

**Zadania egzaminacyjne są wydrukowane
na następnych stronach.**

Zadanie 1. Rekurencja

Dana jest zdefiniowana rekurencyjnie funkcja $A(m, n)$, gdzie m i n są dodatnimi liczbami całkowitymi.

$$A(m, n) = \begin{cases} m & \text{gdy } n = 1 \\ A\left(2 \cdot m, \frac{n}{2}\right) & \text{gdy } n > 1 \text{ oraz } n \text{ jest podzielne przez } 2 \\ 2 \cdot A\left(m, \frac{n-1}{2}\right) + m & \text{gdy } n > 1 \text{ oraz } n \text{ nie jest podzielne przez } 2 \end{cases}$$

1.1.

0-1-
2-3

Zadanie 1.1. (0-3)

Obliczenie wartości funkcji $A(3, 9)$ wprost z definicji wymaga trzech wywołań rekurencyjnych: $A(3, 4)$, $A(6, 2)$, $A(12, 1)$, ponieważ:

$$A(3, 9) = 2 \cdot A(3, 4) + 3 = 2 \cdot A(6, 2) + 3 = 2 \cdot A(12, 1) + 3 = 2 \cdot 12 + 3 = 27$$

Uzupełnij poniższą tabelę. Podaj liczbę wywołań rekurencyjnych funkcji A oraz wypisz wywołania rekurencyjne wraz z ich argumentami (w ostatnim wierszu podaj tylko liczbę wywołań rekurencyjnych).

m	n	liczba wywołań rekurencyjnych funkcji A	wywołania rekurencyjne funkcji A
3	9	3	$A(3, 4)$, $A(6, 2)$, $A(12, 1)$
2^5	2^5		
10	15		
1	$2^{100} + 1$		

Miejsce na obliczenia (brudnopis)



1.2.

0-1

10

Miejsce na obliczenia (brudnopis)

[illegible]

1.3.

0-1-2-3

Zadanie 1.3. (0–3)

Uzupełnij tabelę. W drugiej kolumnie podaj liczbę wywołań rekurencyjnych funkcji A dla każdej wartości n podanej w tabeli (drugiego argumentu wywołania funkcji, pierwszy jest nieistotny w tym zadaniu). W trzeciej kolumnie podaj wyrażenie, którego wartość jest równa drugiemu argumentowi funkcji w i -tym wywołaniu rekurencyjnym dla wszystkich wartości i większych bądź równych 1 i mniejszych bądź równych całkowitej liczbie wywołań.

n – drugi argument wywołania funkcji	liczba wywołań rekurencyjnych	wartość drugiego argumentu A w i -tym wywołaniu rekurencyjnym
8	3	$\frac{8}{2^i}$ (lub 2^{3-i})
2^k		
$2^k - 1$		

gdzie k jest pewną liczbą całkowitą dodatnią większą od 2.

Miejsce na obliczenia (brudnopis)



Rozważamy dodawanie pisemne dwóch liczb zapisanych w systemie dziesiętnym, zilustrowane na przykładzie.

<i>Przeniesienie:</i>	1	1	1		1	
Liczba <i>a</i> :		2	7	7	3	2
Liczba <i>b</i> :	+	7	2	6	1	9
		<u>1</u>	<u>0</u>	<u>0</u>	<u>3</u>	<u>5</u>
		1	0	0	3	5

W tym przykładzie mamy 4 przeniesienia.

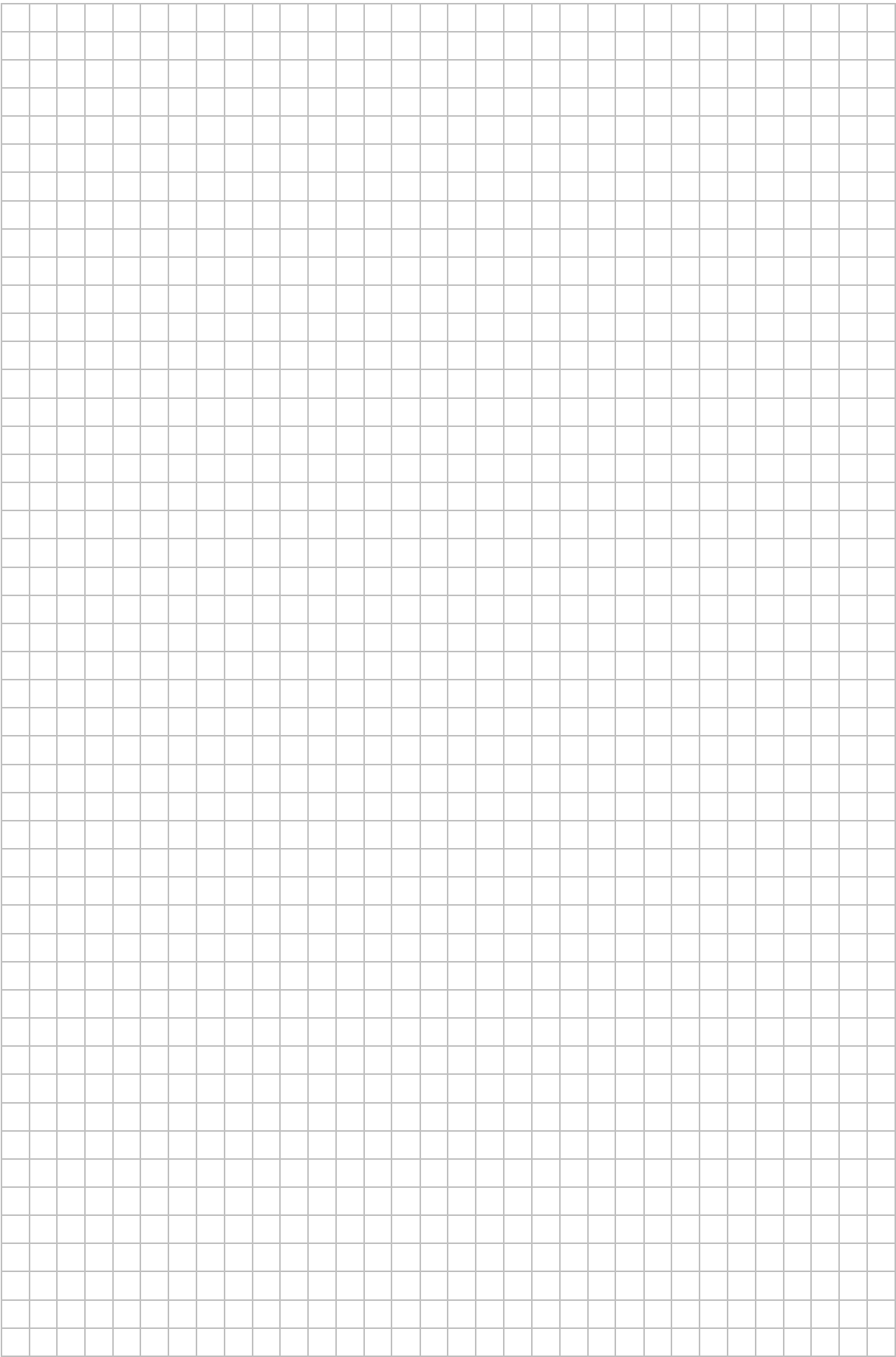
Zadanie 2.1. (0–1)

Dla danych dwóch liczb: a i b , podaj, ile razy pojawiają się przeniesienia podczas ich dodawania.

Liczba a	Liczba b	Liczba przeniesień
37932	12528	3
88765	11111	
456789	222222	

Miejsce na obliczenia (brudnopis)

This image shows a full page of blank graph paper. The grid consists of thin, light gray horizontal and vertical lines that intersect to form a uniform pattern of small squares across the entire surface. There are no margins, text, or other markings present.



Zadanie 3. Pary słów

W pliku tekstowym `pary.txt` znajduje się 500 par słów złożonych z liter alfabetu angielskiego *a, b, ..., z*. Każda para słów jest zapisana w osobnym wierszu. Słowa w wierszu są oddzielone pojedynczym odstępem, a długość każdego z nich nie przekracza 50 znaków.

Pierwszych pięć wierszy pliku `pary.txt` zawiera następujące pary słów:

```
bcba babb
abaa ccc
bcb abbba
bca cdd
aadc ddcddccaba
```

Napisz program (lub kilka programów), który(-e) znajdzie(-ą) i da(dzą) odpowiedzi do podanych zadań. Odpowiedzi do poszczególnych zadań zapisz w pliku `wyniki3.txt`. Każdą odpowiedź poprzedź numerem oznaczającym zadanie.

Do dyspozycji masz plik `pary_przyklad.txt`, który zawiera 500 par słów. Odpowiedzi dla tego pliku podano w treściach zadań. Możesz sprawdzać na nim działanie swojego programu.

Uwaga: Pamiętaj, że Twój program musi ostatecznie działać na pliku `pary.txt`.

3.1.

0-1-2

Zadanie 3.1. (0-2)

Niech $f(s)$ oznacza sumę kodów ASCII znaków występujących w słowie s . Podaj parę słów s_1, s_2 występujących w jednym wierszu pliku `pary.txt`, dla których wartość $|f(s_1) - f(s_2)|$ (wartość bezwzględna różnicy sum kodów ASCII) jest największa, oraz podaj tę wartość. Jest tylko jedna taka para słów w pliku.

Przykład:

Dla pary słów `oko` i `pies`, mamy następujące wartości

$$f(\text{oko}) = 111 + 107 + 111 = 329$$

$$f(\text{pies}) = 112 + 105 + 101 + 115 = 433$$

$$\text{oraz } |f(\text{oko}) - f(\text{pies})| = 104.$$

Dla pliku `pary_przyklad.txt` poprawną odpowiedzią jest
`eddcdd eededcddceeeededcc 1403`

3.2.

0-1-
2-3

Zadanie 3.2. (0-3)

Wspólną liczbę wystąpień litery x w słowach s_1, s_2 oznaczmy przez $W(x, s_1, s_2)$ i definiujemy jako

$$W(x, s_1, s_2) = \text{minimum}(d(x, s_1), d(x, s_2))$$

gdzie $d(x, s)$ oznacza liczbę wystąpień litery x w słowie s .

Podaj parę słów występujących w jednym wierszu w pliku `pary.txt`, dla której suma wspólnych wystąpień wszystkich liter jest największa, oraz podaj tę sumę. Jest jedna taka para.



Przykład:

Dla poniższej pary słów wypisano wspólną liczbę wystąpień wszystkich liter (wyniki wypisano w kolejności alfabetycznej):

adabbcdde aadabbbccdc → a: 2, b: 2, c: 1, d: 2, e: 0

Suma wspólnych wystąpień wszystkich liter dla pary z tego przykładu wynosi:

$2 + 2 + 1 + 2 = 7$.

Dla pliku `pary_przyklad.txt` poprawną odpowiedzią jest

ccccceddddecde ccdcddecc 10

Zadanie 3.3. (0–4)

Prefiksosufiksem pary słów s_1 , s_2 nazywamy słowo, które jest początkiem s_1 (czyli s_1 zaczyna się tym słowem) oraz końcem s_2 (czyli s_2 kończy się tym słowem) lub początkiem s_2 oraz końcem s_1 .

Podaj wszystkie pary słów z pliku `pary.txt`, dla których **najdłuższy** prefiksosufiks ma **co najmniej 5** liter. Dla każdej podanej w odpowiedzi pary słów podaj długość najdłuższego prefiksosufiksu tej pary.

Przykłady:

Dla pary `aabbbca caacaab` mamy następujące prefiksosufiksy:

- `aab` – początek pierwszego słowa i koniec drugiego
- `ca` – początek drugiego słowa i koniec pierwszego.

Najdłuższy prefiksosufiks ma długość 3, zatem para tych słów nie spełnia wymaganych warunków.

Dla pary `abbaabaa baabaabba` mamy następujące prefiksosufiksy:

- `a` – początek pierwszego słowa i koniec drugiego
- `abba` – początek pierwszego słowa i koniec drugiego
- `baa` – początek drugiego słowa i koniec pierwszego
- `baabaa` – początek drugiego słowa i koniec pierwszego.

Najdłuższy prefiksosufiks ma długość 6, zatem para spełnia warunki wymagane w zadaniu.

Dla pliku `pary_przyklad.txt` poprawną odpowiedzią jest

`ecedddeed ddeedd 6`

(najdłuższy prefiksosufiks ma długość 6)

Do oceny oddajesz:

- plik tekstowy `wyniki3.txt` zawierający odpowiedzi do zadań 3.1.–3.3. (odpowiedź do każdego zadania powinna być poprzedzona jego numerem)
- plik(-i) zawierający(-e) kody źródłowe Twoich programów o nazwach odpowiednio (uwaga: brak tych plików jest równoznaczny z brakiem rozwiązania zadania):

zadanie 3.1.

zadanie 3.2.

zadanie 3.3.

3.3.

0–1–

2–3–4

Zadanie 4. Korporacja

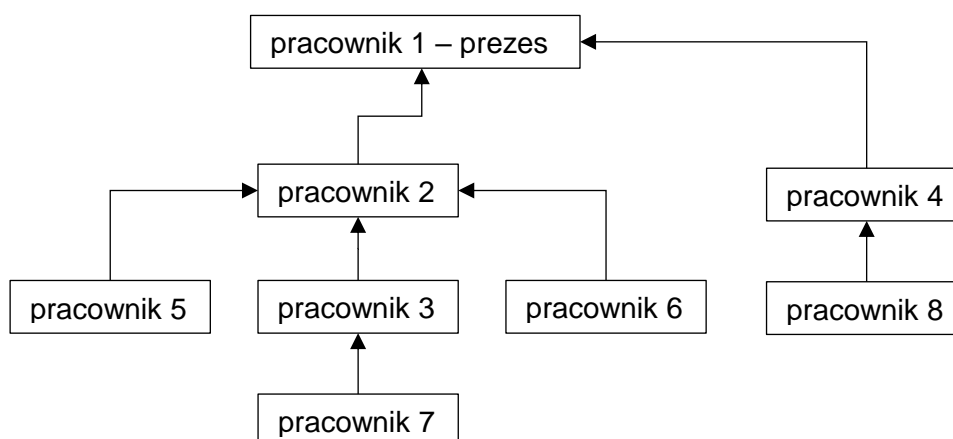
W korporacji pracuje n osób, które na potrzeby zadania ponumerujemy liczbami $1, 2, \dots, n$. Pracownik numer 1 jest prezesem korporacji, a każdy z pozostałych pracowników ma dokładnie jednego bezpośredniego przełożonego.

Numer bezpośredniego przełożonego pracownika x jest zawsze mniejszy od numeru tego pracownika.

Prezes korporacji nie ma żadnego przełożonego. Przełożonym pracownika jest jego bezpośredni przełożony i każdy przełożony tego bezpośredniego przełożonego.

Jeśli x jest (bezpośrednim) przełożonym y , to powiemy, że y jest (bezpośrednim) podwładnym x . Prezes jest przełożonym każdego pracownika.

Przykład 1.



Rysunek 1. Przykład hierarchii w korporacji (strzałki wskazują na bezpośrednich przełożonych).

Przełożonymi pracownika 7 są pracownicy 3, 2 i 1. Bezpośrednim przełożonym pracownika 7 jest pracownik 3. Pracownik 3 (podobnie jak 5 oraz 6) jest bezpośrednim podwładnym pracownika 2, a podwładnymi pracownika 2 są pracownicy 3, 5, 6 i 7.

4.1.

0–1

Zadanie 4.1. (0–1)

Dla każdego pracownika z przykładowej hierarchii w korporacji (Rysunek 1.) określ, ilu ma on bezpośrednich podwładnych oraz ilu ma wszystkich podwładnych.

Numer pracownika	Liczba bezpośrednich podwładnych	Liczba wszystkich podwładnych
1		
2	3	4
3		
4		
5		
6		
7		
8	0	0

Informacja do zadań 4.2.–4.4.

Dany jest plik `korpo.txt` zawierający $n = 50\,000$ liczb, który opisuje strukturę korporacji. W pierwszym wierszu jest liczba 0 oznaczająca brak przełożonego dla prezesa korporacji, który ma numer 1. W i -tym ($2 \leq i \leq 50\,000$) wierszu pliku jest numer pracownika, który jest bezpośrednim przełożonym pracownika i .

Przykład 2.

Zapis w pliku dla przykładowej korporacji (Rysunek 1) miałby postać:

```
0
1
2
1
2
2
3
4
```

Napisz program(-y), który(-e) znajdzie(-ą) i da(dzą) odpowiedzi do podanych zadań. Odpowiedzi zapisz w pliku `wyniki4.txt`, a każdą z nich poprzedź numerem odpowiedniego zadania.

Do Twojej dyspozycji jest również plik `korpo_przyklad.txt`, w identycznym formacie, w którym bezpośrednim przełożonym pracowników 2 i 3 jest pracownik 1, a wszystkich innych – pracownik 3. Odpowiedzi dla danych z tego pliku są podane pod treściami zadań.

Uwaga: Pamiętaj, że Twój program musi ostatecznie działać dla pliku `korpo.txt`.

Zadanie 4.2. (0–2)

Na podstawie danych zapisanych w pliku `korpo.txt` podaj, ilu pracowników nie jest przełożonym żadnego pracownika.

Dla danych z pliku `korpo_przyklad.txt` poprawna odpowiedź to:
49 998

4.2.

0–1–2

Zadanie 4.3. (0–2)

Podaj, który pracownik spośród zapisanych w pliku `korpo.txt` ma najwięcej bezpośrednich podwładnych. Podaj numer tego pracownika oraz liczbę jego bezpośrednich podwładnych.

Dla danych z pliku `korpo_przyklad.txt` poprawna odpowiedź to:
3 49 997
(pracownik numer 3 ma 49 997 bezpośrednich podwładnych).

4.3.

0–1–2

4.4.

0-1-
2-3**Zadanie 4.4. (0–3)**

Policz, ilu najwięcej przełożonych ma jeden pracownik. Podaj tę liczbę oraz podaj, ilu pracowników ma taką liczbę przełożonych.

Dla danych z pliku `korpo_przyklad.txt` poprawna odpowiedź to:

2 49 997

(największa liczba przełożonych: 2, liczba pracowników, którzy mają 2 przełożonych: 49 997).

Do oceny oddajesz:

- plik `wyniki4.txt` – zawierający odpowiedzi do zadań 4.2.–4.4.
(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 4.2.

zadanie 4.3.

zadanie 4.4.

Zadanie 7. Staw

Pan Iksiński stał się właścicielem stawu o powierzchni całkowitej 10 000 m². Pierwszą rośliną, którą postanowił w nim umieścić, jest rzęsa wodna, która osiąga wysoką skuteczność rozmnażania wegetatywnego, tzn. przy odpowiedniej temperaturze i wielkości opadów potrafi znacząco zwiększyć rozmiar zajmowanej powierzchni.

W pliku `staw.txt` są zawarte następujące informacje, rozdzielone znakami tabulacji:

Data – data pomiaru

Temp – temperatura w danym dniu w °C, zaokrąglona do jednego miejsca po przecinku

Opady – wielkość opadu w mm, zaokrąglona do liczby całkowitej.

Przykład:

Data	Temp	Opady
2022-01-01	9,8	5
2022-01-02	8,1	1
2022-01-03	9	2
2022-01-04	7,3	1
2022-01-05	5,7	5

Plik zawiera dane z całego 2022 roku.

Z wykorzystaniem danych zawartych w pliku oraz dostępnych narzędzi informatycznych wykonaj podane zadania. Wyniki zapisz w pliku tekstowym `wyniki7.txt`. Odpowiedź do każdego zadania poprzedź numerem tego zadania.

7.1.

0–1–
2–3

Zadanie 7.1. (0–3)

Utwórz zestawienie średnich miesięcznych temperatur w zaokrągleniu do jednego miejsca po przecinku. Na podstawie wykonanego zestawienia utwórz wykres kolumnowy, porównujący te wartości. Pamiętaj o czytelnym opisie wykresu (tytuł wykresu, opisy osi, oznaczenie miesięcy na osi X).

7.2.

0–1–2

Zadanie 7.2. (0–2)

Dla każdego miesiąca wyznacz długość najdłuższego ciągu kolejnych dni bez opadów w tym miesiącu (wartość opadów w tych dniach jest równa 0).



Informacja do zadań 7.3.–7.4.

Na potrzeby zadania przyjmujemy, że w kolejnym roku przez 184 dni, od 1 marca 2023 do 31 sierpnia 2023, temperatury i opady utrzymywały się na stałym poziomie, co pozwalało na regularny wzrost rzęsy wodnej w tempie rozrostu 1,75% dziennie. Przyrost rzęsy następował w nocy, a pomiar zarośnięcia stawu – rano.

1 marca 2023 rano staw był zarośnięty rzęsą w 20%, tj. rzęsa wodna zajmowała 2000 m^2 . W związku z tym, że staw nie powinien być zarośnięty w całości, właściciel postanowił pozbywać się jej nadmiaru. Do zbiornika wpuścił 80 amurów białych, z których każdy zjadał w ciągu dnia $0,25 \text{ m}^2$ rzęsy wodnej. Dodatkowo co piątek w ciągu dnia odławiał 60 m^2 rzęsy wodnej.

Uwaga: 30 kwietnia rano staw był zarośnięty w 25,79%.

Zadanie 7.3. (0–2)

Podaj, w którym dniu (licząc od 1 marca 2023) pomiar wykazał, że rzęsa wodna po raz pierwszy zajęła więcej niż 75% procent powierzchni stawu.

7.3.

0–1–2

Zadanie 7.4. (0–2)

Podaj, jaka jest najmniejsza liczba amurów białych, jaką musi wpuścić właściciel, by rzęsa wodna w całym badanym okresie zajmowała maksymalnie 50% powierzchni stawu.

7.4.

0–1–2

Do oceny oddajesz:

- plik tekstowy `wyniki7.txt` zawierający odpowiedzi do zadań 7.1.–7.4. Odpowiedź do każdego zadania powinna być poprzedzona jego numerem.
- plik zawierający wykres do zadania 7.1. o nazwie:
.....
- plik(pliki) zawierający(-e) komputerową realizację Twoich rozwiązań o nazwie(-ach) (uwaga: brak tych plików jest równoznaczny z brakiem rozwiązania zadania):
.....

Zadanie 8. Sieć sklepów

W trzech plikach tekstowych o nazwach `klienci.txt`, `transakcje.txt`, `opis_transakcji.txt` zapisano dane o sprzedaży towarów w pewnej sieci sklepów w porach wieczornych pierwszych dni miesiąca. Dane obejmują informacje od 1. do 3. dnia miesiąca w miesiącach od stycznia do czerwca 2025 roku w godzinach od 22:00 do 23:59. Pierwszy wiersz każdego z plików jest wierszem nagłówkowym, a dane w wierszach rozdzielono tabulatorami.

Plik `klienci.txt` zawiera informacje na temat zarejestrowanych klientów sieci sklepów.

W każdym wierszu zapisano:

`IdKlienta` – unikatowy identyfikator klienta (liczba całkowita)

`Imie` – imię klienta (tekst do 20 znaków)

`Nazwisko` – nazwisko klienta (tekst do 20 znaków)

`Plec` – płeć klienta (znak K lub M).

Przykład:

<code>IdKlienta</code>	<code>Imie</code>	<code>Nazwisko</code>	<code>Plec</code>
1	Zofia	Sawicka	K
2	Zuzanna	Chmiel	K
3	Hanna	Sokolowska	K

Plik `transakcje.txt` zawiera informacje o dokonanych transakcjach. W każdym wierszu zapisano:

`IdTransakcji` – unikatowy identyfikator transakcji (uwaga: identyfikatory nie są nadane chronologicznie – nie są kolejnymi numerami transakcji)

`DataTransakcji` – datę transakcji (dd.mm.rrrr)

`IdKlienta` – identyfikator klienta, który dokonał transakcji

`IdSklepu` – identyfikator sklepu, w którym dokonano transakcji

`IdSprzedawcy` – identyfikator sprzedawcy, który obsługiwał klienta przy kasie; jeśli klient korzystał z **kasy samoobsługowej**, pole jest **puste (brak danych)**.

Przykład:

<code>IdTransakcji</code>	<code>DataTransakcji</code>	<code>IdKlienta</code>	<code>IdSklepu</code>	<code>IdSprzedawcy</code>
1	01.01.2025	1814	12	160
2	01.02.2025	550	4	125
3	01.03.2025	1408	16	65
4	01.04.2025	1146	14	33
5	01.05.2025	630	9	
6	01.06.2025	1011	12	120

Plik o nazwie `opis_transakcji.txt` zawiera informacje o produktach zakupionych w danej transakcji. W każdym wierszu zapisano:

`IdTransakcji` – identyfikator transakcji (uwaga: identyfikatory mogą się powtarzać)

`IdProduktu` – identyfikator zakupionego produktu

`Cena` – cena jednostkowa zakupionego produktu

`Liczba` – liczba określająca, ile sztuk danego produktu zakupiono.



Przykład:

IdTransakcji	IdProduktu	Cena	Liczba
485	78	5,66	11
104	87	7	4
104	84	2	5
246	42	7,0	15

Z wykorzystaniem danych zawartych w podanych plikach oraz dostępnych narzędzi informatycznych podaj odpowiedzi do zadań 8.1.–8.4. Odpowiedzi zapisz w pliku `wyniki8.txt`, a każdą z nich poprzedź numerem odpowiedniego zadania.

Zadanie 8.1. (0–1)

Podaj imię i nazwisko klienta, który dokonał łącznie najwięcej transakcji w całym analizowanym okresie.

8.1.

0–1

Zadanie 8.2. (0–2)

Podaj, ile kobiet (K) oraz ilu mężczyzn (M) spośród klientów sieci sklepów nie kupiło niczego w całym analizowanym okresie.

8.2.

0–1–2

Zadanie 8.3. (0–2)

Podaj liczbę różnych sklepów, w których dokonano transakcji w kasach samoobsługowych, oraz podaj, ile zapłacono łącznie za zakupy w tych kasach.

8.3.

0–1–2

Zadanie 8.4. (0–2)

Niektórzy sprzedawcy pracowali w różnych sklepach sieci w ciągu miesiąca.

Podaj `IdSprzedawcy`, który obsługiwał klientów w największej liczbie różnych sklepów w jednym miesiącu, oraz podaj ten miesiąc (nazwę lub numer).

8.4.

0–1–2

Do oceny oddajesz:

- plik tekstowy `wyniki8.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):

.....

Zadanie 8.5. (0–2)

Do bazy danych utworzonej na podstawie opisanych wcześniej plików dodano dwie kolejne tabele: o nazwie *Produkty* i o nazwie *Kategorie*, w których zapisano informacje o sprzedawanych produktach.

Tabela *Kategorie* zawiera następujące pola:

IdKategorii – unikatowy identyfikator kategorii produktu

NazwaKategorii – nazwa kategorii produktu.

Tabela *Produkty* zawiera następujące pola:

IdProduktu – unikatowy identyfikator produktu

`IdKategorii` – identyfikator kategorii, do której należy dany produkt

Nazwa – nazwa produktu

Opis – opis produktu.

Napisz w języku SQL zapytanie, w wyniku którego zostaną wypisane wszystkie zakupione produkty (IdProduktu i Nazwa) z kategorii (NazwaKategorii) „spozywcze” zawierające w opisie fragment: „do ekspresu kolbowego”.

Miejsce na zapis zapytania

A full-page view of a blank sheet of graph paper. The grid consists of thin, light gray horizontal and vertical lines forming small squares across the entire page. There are no margins, text, or other markings on the paper.

BRUDNOPIS (*nie podlega ocenie*)

INFORMATYKA

Poziom rozszerzony

Formuła 2023



INFORMATYKA

Poziom rozszerzony

Formuła 2023



INFORMATYKA

Poziom rozszerzony

Formuła 2023

