



WIESŁAWA SUCHOCKA



INFORMATYKA

ZBIÓR ZADAŃ

DLA POCZĄTKUJĄCYCH PROGRAMISTÓW

Publikacja powstała w ramach realizacji projektu SZKOŁA JUTRA
w IV Liceum Ogólnokształcącym im. Cypriana Kamila Norwida w Białymstoku

OD AUTORKI

„Zbiór zadań dla początkujących programistów” jest dopełnieniem ćwiczeń omawianych na lekcjach informatyki. Pozwala kształcić i na wiele sposobów utrwalać treści opisane w podstawie programowej.

Nie mając gotowych odpowiedzi uczeń jest zmuszony do samodzielnych poszukiwań. Cennym źródłem wiedzy stają się wówczas bardziej doświadczeni koledzy czy Internet, ale przede wszystkim **zachęcam do dodatkowych konsultacji z nauczycielem.**

NA DOBRY POCZĄTEK ... KILKA WSKAZÓWEK

- ☒ Zwracaj uwagę na **specyfikację zadania**:
 - Standardowe wejście oznacza klawiaturę, a standardowe wyjście - ekran komputera.
 - Jeżeli na wyjściu tekst ma być wypisany wielkimi literami (TAK), to wszelkie zapisy typu: „tak”, „Tak”, itp. uznajemy jako nie zgodne ze specyfikacją.
 - Nie sprawdzaj przypadków dla danych wejściowych, które nie są zgodne ze specyfikacją.
 - Jeżeli liczba, jako dana wejściowa, nie mieści się w zakresie znanych Ci typów liczbowych (np. 10^{200}) zastosuj typ string.
- ☒ Pamiętaj, że dla liczb całkowitych zapis x/y oznacza **część całkowitą z dzielenia** liczby x przez y , więc zarówno zapis $6/2$ jak i $7/2$ generuje wynik równy 3.
- ☒ Jeżeli we wzorze na obliczanie pola koła wstawiasz przybliżenie np. $\pi = 3.14$ narażasz się na błąd związany z wymaganą dokładnością wyniku. Można zastosować w programie stałą **M_PI**, która znajduje się w bibliotece `<cmath>`
- ☒ **Wykorzystuj sprawdzarkę** umieszczoną na szkolnym serwerze – to serwis automatycznie oceniający Twoje rozwiązania.
- ☒ Staraj się dbać o **jak najlepszą efektywność** pisanych przez Ciebie programów.

Zacznij pracę już teraz. Bądź systematyczny.

Najlepszy programista też był kiedyś początkującym informatykiem.



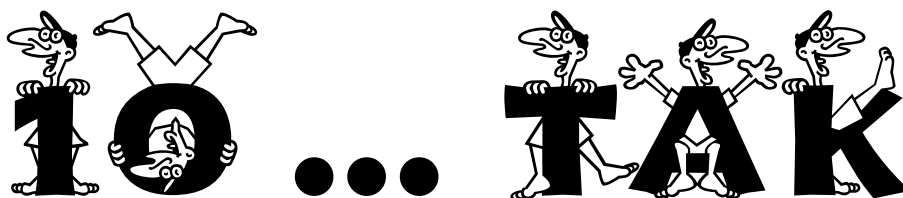
1. ZACZYNAMY ... OD PODSTAW (C++)

*Jaki typ danych zadeklarować w zadaniu? Czy zastosować pętlę FOR czy WHILE?
Sprawdź, czy znasz odpowiedzi na te pytania rozwiązując zadania z pierwszego rozdziału.*

1.1. Napisz program, który sprawdzi, czy liczba jest dwucyfrowa.

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba naturalna n ($0 \leq n \leq 10^9$)

Wyjście: Na standardowe wyjście należy wypisać słowo TAK, jeśli podana liczba jest dwucyfrowa, NIE – w przeciwnym wypadku.



1.2. Napisz program, który sprawdzi, czy liczba jest nieparzysta.

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n ($-10^9 \leq n \leq 10^9$)

Wyjście: Na standardowe wyjście należy wypisać słowo TAK, jeśli podana liczba jest nieparzysta, NIE – w przeciwnym wypadku.

1.3. Napisz program, który sprawdzi, czy liczba jest podzielna przez 10.

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba naturalna n ($0 \leq n \leq 10^{200}$)

Wyjście: Na standardowe wyjście należy wypisać słowo TAK, jeśli podana liczba jest podzielna przez 10, NIE – w przeciwnym wypadku.

1.4. Napisz program, który sprawdzi, czy liczba jest podzielna przez 4

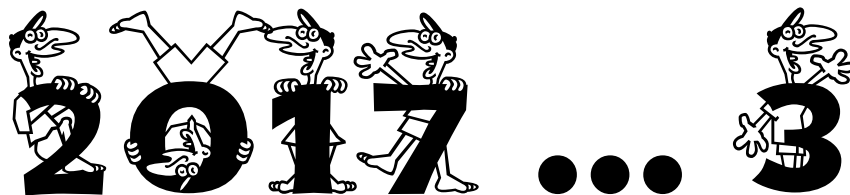
Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba naturalna n ($0 \leq n \leq 10^{200}$)

Wyjście: Na standardowe wyjście należy wypisać słowo TAK, jeśli podana liczba jest podzielna przez 4, NIE – w przeciwnym wypadku.

1.5. Napisz program, który obliczy ostatnią cyfrę liczby 3^n

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba naturalna n , $n \leq 10^9$

Wyjście: Na standardowe wyjście należy wypisać jedną liczbę – ostatnią cyfrę wyniku obliczeń 3^n .



1.6. Napisz program, który obliczy objętość i pole powierzchni całkowitej prostopadłościanu o podanych wymiarach

Wejście: W pierwszym wierszu standardowego wejścia znajdują się trzy liczby rzeczywiste a, b, c , gdzie $0 < a, b, c \leq 1000$.

Wyjście: Na standardowe wyjście należy wypisać dwie liczby oznaczające odpowiednio objętość i pole powierzchni całkowitej prostopadłościanu o krawędziach długości a, b, c . Wyniki podajemy z dokładnością do 2 miejsc po przecinku.

1.7. Napisz program, który obliczy pole i obwód koła o podanym promieniu.

Wejście: liczba rzeczywista r , oznaczająca długość promienia koła ($0 < r \leq 1000$).

Wyjście: W pierwszym wierszu standardowego wyjścia należy wypisać długość obwodu koła o promieniu r , a w drugim wierszu - pole tego koła. Oba wyniki wypisujemy z dokładnością do 3 miejsc po przecinku.

Przykład:

Dla danych wejściowych:

6

Poprawnym wynikiem jest:

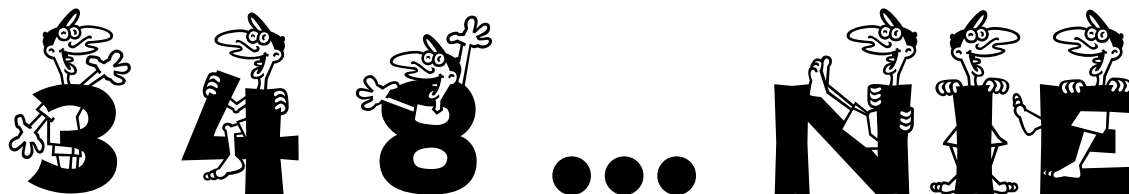
37.699

113.097

1.8. Napisz program, który sprawdzi, czy 3 podane liczby mogą być długościami boków trójkąta.

Wejście: W pierwszym wierszu standardowego wejścia znajdują się trzy liczby rzeczywiste a, b, c , gdzie $0 < a, b, c \leq 1000$

Wyjście: Na standardowe wyjście należy wypisać słowo TAK, jeśli z podanych długości można zbudować trójkąt, NIE – w przeciwnym wypadku.



1.9. Napisz program, który znajdzie najmniejszą spośród czterech liczb

Wejście: W pierwszym wierszu standardowego wejścia znajdują się cztery różne liczby całkowite a, b, c, d , gdzie $-1000000 < a, b, c, d \leq 1000000$

Wyjście: Na standardowe wyjście należy wypisać najmniejszą z podanych liczb.

1.10. Napisz program, który sprawdzi, czy dla podanych czterech liczb spełnione jest równanie $\frac{a}{b} = \frac{c}{d}$

Wejście: W pierwszym wierszu standardowego wejścia znajdują się cztery liczby całkowite a, b, c, d , gdzie $0 < a, b, c, d \leq 1000000$

Wyjście: Na standardowe wyjście należy wypisać słowo TAK, jeśli spełniona jest równość $\frac{a}{b} = \frac{c}{d}$, NIE – w przeciwnym wypadku.

1.11. Napisz program, który obliczy wartość bezwzględną różnicy liczb a i b

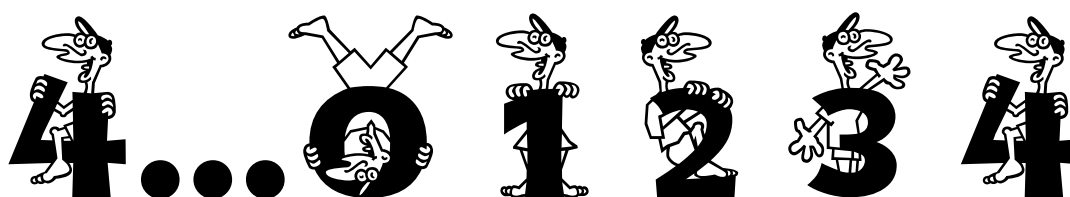
Wejście: W pierwszym wierszu standardowego wejścia znajdują się dwie liczby całkowite a, b , gdzie $0 < a, b \leq 1000000$

Wyjście: Na standardowe wyjście należy wypisać wynik działania $|a - b|$

1.12. Napisz program, który wypisze na ekranie kolejne liczby naturalne nie większe od n

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n , gdzie $0 \leq n \leq 1000$

Wyjście: Na standardowe wyjście należy wypisać po spacji kolejne liczby naturalne nie większe od n .



1.13. Napisz program, który wypisze na ekranie kolejne liczby naturalne z podanego przedziału $<a, b>$ zaczynając od największej.

Wejście: W pierwszym wierszu standardowego wejścia znajdują się dwie liczby naturalne a i b , gdzie $0 \leq a < b \leq 1000$

Wyjście: Na standardowe wyjście należy wypisać po spacji kolejne liczby naturalne od b do a .

Przykład. Dla danych wejściowych: 3 9

Poprawną odpowiedzią jest: 9 8 7 6 5 4 3

1.14. Napisz program, który wypisze na ekranie wszystkie liczby dwucyfrowe, które dzielą się przez 6.

Wyjście: Na standardowe wyjście należy wypisać po spacji kolejne liczby naturalne dwucyfrowe podzielne przez 6.

1.15. Napisz program, który obliczy sumę wszystkich liczb naturalnych nie większych n .

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n , $0 \leq n \leq 10^6$

Wyjście: Na standardowe wyjście należy wypisać jedną liczbę oznaczającą sumę wszystkich liczb naturalnych nie większych od n .

1.16. Napisz program, który wypisze na ekranie sumę wszystkich liczb czterocyfrowych.

Wyjście: Na standardowe wyjście należy wypisać wynik dodawania:

$$1000 + 1001 + 1002 + \dots + 9999$$

1.17. Napisz program, który wypisze na ekranie 50 liczb postaci: -19 -15 -11 -7

Wyjście: Na standardowe wyjście należy wypisać po spacji dokładnie 50 liczb zaczynając od liczby (-19) zgodnie z poleceniem.

1.18. Napisz program, który obliczy średnią arytmetyczną liczb całkowitych z podanego przedziału $< a, b >$

Wejście: W pierwszym wierszu standardowego wejścia znajdują się dwie liczby całkowite a, b , gdzie $0 \leq a < b \leq 1000000$

Wyjście: Na standardowe wyjście należy wypisać jedną liczbę – średnią arytmetyczną wszystkich liczb z przedziału $< a, b >$

1.19. Napisz program, który obliczy ile jest liczb parzystych w podanym przedziale $< a, b >$

Wejście: W pierwszym wierszu standardowego wejścia znajdują się dwie liczby całkowite a i b , gdzie $a < b$ oraz $-1000000 \leq a, b \leq 1000000$

Wyjście: Na standardowe wyjście należy wypisać jedną liczbę oznaczającą ilość liczb parzystych z przedziału $< a, b >$.



1.20. Napisz program, który obliczy ilość wystąpień podanej liczby x .

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n , $n \leq 100$ oznaczająca ile liczb mamy do sprawdzenia.

W drugim wierszu wejścia znajduje się n cyfr.

W trzecim wierszu wejścia określamy szukaną liczbę x

Wyjście: Na standardowe wyjście należy wypisać jedną liczbę oznaczającą ilość wystąpień liczby x .

Przykład:

Dla danych wejściowych:

7

2 4 6 4 3 7 4

4

Poprawną odpowiedzią jest: 3

1.21. Napisz program, który obliczy ile w podanym przedziale $\langle a, b \rangle$ jest liczb, których ostatnia cyfra jest liczbą pierwszą.

Wejście: W pierwszym wierszu standardowego wejścia znajdują się dwie liczby całkowite a i b , gdzie $a < b$ oraz $0 \leq a, b \leq 10^9$

Wyjście: Na standardowe wyjście należy wypisać jedną liczbę oznaczającą ilość liczb z przedziału $\langle a, b \rangle$, w których ostatnia cyfra jest liczbą pierwszą.

1.22. Napisz program, który obliczy sumę cyfr podanej liczby naturalnej n .

Wejście: W pierwszym wierszu standardowego wejścia znajduje się liczba naturalna $n \leq 10^9$

Wyjście: Na standardowe wyjście należy wypisać liczbę, która jest sumą cyfr liczby n .

1.23. Napisz program drukujący na ekranie liczby/gwiazdki zgodnie z poniższym schematem.

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n – liczba wierszy. W przykładzie wynik dla $n = 5$

Wyjście:

1

1 2

1 2 3

1 2 3 4

1 2 3 4 5

1.24. Napisz program drukujący na ekranie liczby zgodnie z poniższym schematem.

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n – liczba wierszy. W przykładzie wynik dla $n = 5$

Wyjście:

```
1
2 4
3 6 9
4 8 12 16
5 10 15 20 25
```

1.25. Napisz program drukujący na ekranie liczby zgodnie z poniższym schematem.

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n – liczba wierszy. W przykładzie wynik dla $n = 5$

Wyjście:

```
2
4 5
8 9 10
16 17 18 19
32 33 34 35 36
```

1.26. Wygeneruj tablicę o wymiarach 6×6 , której elementy są równe sumie indeksów wiersza i kolumny. Wypisz elementy tablicy z podziałem na wiersze i kolumny.

Napisz program (programy), który wykonuje następujące polecenia:

- Wyzeruj elementy znajdujące się na głównej przekątnej
- Zamień zawartość kolumny o indeksie 0 z kolumną o indeksie 5, czyli kolumnę pierwszą z ostatnią.

WYPISZ NOWĄ ZAWARTOŚĆ TABLICY {po zrobieniu poleceń (a) i (b)}

- Wszystkie elementy tablicy równe 0 zamień na (-5)

WYPISZ NOWĄ ZAWARTOŚĆ TABLICY {po zrobieniu dodatkowo polecenia (c)}

- Oblicz sumę liczb nad główną przekątną
- Oblicz, ile elementów tablicy jest podzielnych przez 3
- Oblicz sumę elementów podzielnych przez 5
- Oblicz sumę tych elementów tablicy, których numer wiersza jest liczbą nieparzystą
- Oblicz, który element tablicy występuje najczęściej. Jeśli jest więcej niż jeden taki element wypisz je wszystkie.

1.27. Napisz program, który wskaże pozycję największej liczby parzystej w podanym zbiorze

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n , $1 \leq n \leq 100$ oznaczająca ilość liczb do sprawdzenia.

W następnej linii znajduje się n liczb naturalnych nie większych od 10^9 oddzielonych spacjami a_i , gdzie $1 \leq i \leq n$.

Wyjście: Na standardowe wyjście należy wypisać jedną liczbę oznaczającą pozycję największej parzystej liczby wśród danych w drugim wierszu. W przypadku kilku takich samych wyników końcowych należy podać najniższą pozycję.

Przykład:

Dla danych wejściowych:

5

4 8 13 8 6

Poprawną odpowiedzią jest: 2

1.28. Napisz program, który dla dwóch podanych liczb całkowitych a i b sprawdzi ile wynosi ich suma ($a + b$), różnica ($a - b$), iloczyn ($a * b$) i wypisze największy z tych wyników wraz z odpowiednim działaniem.

Wejście: W pierwszym i jedynym wierszu standardowego wejścia znajdują się dwie liczby całkowite a i b ($-100 \leq a, b \leq 100$)

Wyjście: Na standardowe wyjście należy wypisać zapis zgodnie z przykładem (liczby ujemne wpisujemy w nawiasie) lub słowo „WIECEJ”, jeśli można uzyskać wynik za pomocą więcej niż jednego działania.

Przykład:

Dla danych wejściowych:

7 - 2

Poprawną odpowiedzią jest: $7 - (-2) = 9$

1.29. Napisz program, który wypisuje poszczególne cyfry podanej liczby naturalnej n .

Wejście: W pierwszym wierszu standardowego wejścia znajduje się liczba naturalna $n \leq 10^9$

Wyjście: Na standardowe wyjście należy wypisać poszczególne cyfry liczby n oddzielone znakiem (-).

Przykład:

Dla danych: 2017

Poprawną odpowiedzią jest: 2-0-1-7

1.30. Napisz program, który wczyta n liczb całkowitych, liczby nieparzyste zamieni na ich sześciiany, a parzyste „skróci” o cyfrę jedności. W przypadku liczby parzystej jednocyfrowej wypisujemy 0.

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n , $0 < n \leq 10000$

Wyjście: Na standardowe wyjście należy wypisać n liczb całkowitych zgodnie z treścią zadania.

Przykład

Dla danych wejściowych:

5

30 4 7 128 3

Poprawnym wynikiem jest: 3 0 343 12 27

1.31. Napisz program, który wyznacza długość najdłuższego słowa w zestawie.

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n , gdzie $1 \leq n \leq 10$. W kolejnych n wierszach znajdują się wyrazy, po jednym w każdym wierszu. Długość słowa nie przekracza 100 znaków.

Wyjście: Na standardowe wyjście należy wypisać długość najdłuższego słowa.

Przykład

Dla danych wejściowych:

4

mama

informatyka

Bartek

dom

Poprawnym wynikiem jest: 11

1.32. Napisz program, który sprawdza, czy podane słowo jest palindromem.

Palindromem nazywamy słowo, które czytane od lewej i od prawej strony jest takie samo.

Przykłady palindromów: AbbA, kajak, sedes.

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedno słowo.

Wyjście: Na standardowe wyjście należy wypisać komunikat TAK – jeżeli słowo jest palindromem, NIE – w przeciwnym wypadku.

1.33. Napisz program, który zamienia każdą literę a w podanym tekście na znak *.

Wejście: W pierwszym wierszu standardowego wejścia znajduje się tekst nie dłuższy niż 250 znaków składający się z małych i wielkich liter alfabetu.

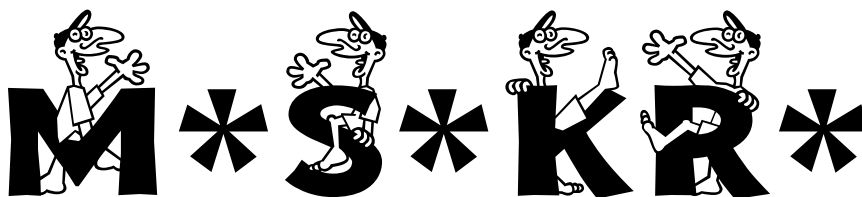
Wyjście: Na standardowe wyjście należy wypisać tekst, w którym każdą literę a zamieniono na *.

Przykład: Dla danych wejściowych:

Ala ma kota

Poprawnym wynikiem jest:

| m* kot*



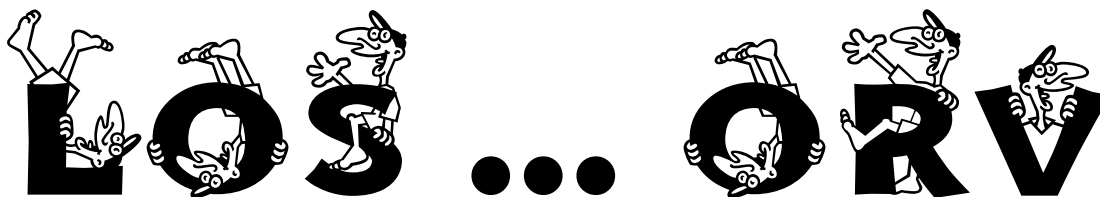
1.34. Napisz program, który każdą literę wyrazu zamienia na literę przesuniętą o 3 pozycje w prawo.

Wejście: W pierwszym wierszu standardowego wejścia znajduje się wyraz składający się tylko z wielkich liter alfabetu angielskiego (o kodach ASCII od 65 do 90).

Wyjście: Na standardowe wyjście należy wypisać tekst, w którym każdą literę zastąpiono cyklicznie literą znajdującą się o 3 miejsca w prawo. Jeżeli przesuwana litera nie ma literowego odpowiednika to alfabet „się zawija” i po literze Z następuje znowu litera A.

Przykład: Dla danych wejściowych: CEZAR

Poprawnym wynikiem jest: FHCDU



1.35. Napisz program, który liczbę naturalną zapisaną w systemie dziesiętnym zamienia na jej reprezentację w systemie dwójkowym.

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n , $0 \leq n \leq 65000$.

Wyjście: Na standardowe wyjście należy wypisać reprezentację binarną liczby n .

1.36. Napisz program, który liczbę naturalną zapisaną w systemie dziesiętnym zamienia na jej reprezentację w systemie p .

Wejście: W pierwszym wierszu standardowego wejścia znajdują się dwie liczby całkowite n i p , $0 \leq n \leq 50000$, $2 \leq p \leq 9$.

Wyjście: Na standardowe wyjście należy wypisać reprezentację liczby n w systemie pozycyjnym o podstawie p .

1.37. Napisz program, który liczbę naturalną zapisaną w systemie p zamienia na jej reprezentację w systemie dziesiętnym.

Wejście: W pierwszym wierszu standardowego wejścia znajdują się dwie liczby całkowite n i p , $2 \leq p \leq 9$, gdzie p jest podstawą systemu, n – liczbą zapisaną w systemie p .

Wyjście: Na standardowe wyjście należy wypisać reprezentację liczby n w systemie dziesiętnym.

1.38. Napisz program, który liczbę naturalną zapisaną w systemie dziesiętnym zamienia na jej reprezentację w systemie szesnastkowym.

Wejście: W pierwszym wierszu standardowego wejścia znajduje się liczba naturalna n , gdzie $0 \leq n \leq 1\,000\,000$ oznaczająca liczbą zapisaną w systemie dziesiętnym.

Wyjście: Na standardowe wyjście należy wypisać reprezentację liczby n w systemie szesnastkowym.

1.39. Napisz program, który liczbę naturalną zapisaną w systemie szesnastkowym zamienia na jej reprezentację w systemie dziesiętnym.

Wejście: W pierwszym wierszu standardowego wejścia znajduje się liczba naturalna n , oznaczająca liczbą zapisaną w systemie szesnastkowym. Liczba n ma co najwyżej 8 cyfr.

Wyjście: Na standardowe wyjście należy wypisać reprezentację liczby n w systemie dziesiętnym.

1.40. Napisz program, który całkowitą liczbę ujemną zapisaną w systemie dziesiętnym zamienia na jej reprezentację w systemie kodowania U2

Wejście: W pierwszym wierszu standardowego wejścia znajduje się liczba naturalna n , gdzie $-10\,000\,000 \leq n < 0$ oznaczająca liczbą zapisaną w systemie dziesiętnym.

Wyjście: Na standardowe wyjście należy wypisać reprezentację liczby n w systemie kodowania U2.

1.41. Napisz program, który ujemną liczbę całkowitą zapisaną w systemie U2 zamienia na jej reprezentację w systemie dziesiętnym.

Wejście: W pierwszym wierszu standardowego wejścia znajduje się liczba n , oznaczająca ujemną liczbę całkowitą zapisaną w systemie kodowania U2. Liczba n ma co najwyżej 16 cyfr.

Wyjście: Na standardowe wyjście należy wypisać wartość liczby n w systemie dziesiętnym.

2.CZTERY MIESIĄCE PÓŹNIEJ ... ALGORYTMY

W poprzednim rozdziale utrwalaliśmy „alfabet C++”, a nawet składaliśmy już „pierwsze zdania” w tym języku. Czas na pisanie „całych wypracowań”, czyli zastosowanie naszych umiejętności do realizacji przykładowych algorytmów zapisanych w podstawie programowej.

UWAGA. Wszystkie pliki do zadań znajdują się na stronie:

<http://4lo.bialystok.pl/> w zakładce: **E-learning**

2.1. LICZBY PARZYSTE

Wejście: W pliku **zad2-1.txt** znajduje się 500 liczb całkowitych n , gdzie $1 \leq n \leq 10^6$, po jednej w każdym wierszu.

Wyjście: Na standardowym wyjściu należy wypisać ilość liczb parzystych znajdujących się w pliku **zad2-1.txt**. Zapisz wynik w systemie dziesiętnym oraz binarnym.

2.2. NAJWIĘKSZA I NAJMNIEJSZA LICZBA W ZBIORZE

Wejście: W pliku **zad2-2.txt** znajduje się 500 liczb całkowitych n , gdzie $1 \leq n \leq 10^6$, po jednej w każdym wierszu.

Wyjście: Na standardowym wyjściu należy wypisać najmniejszą i największą liczbę z pliku **zad2-2.txt**.

2.3. LICZBY PIERWSZE

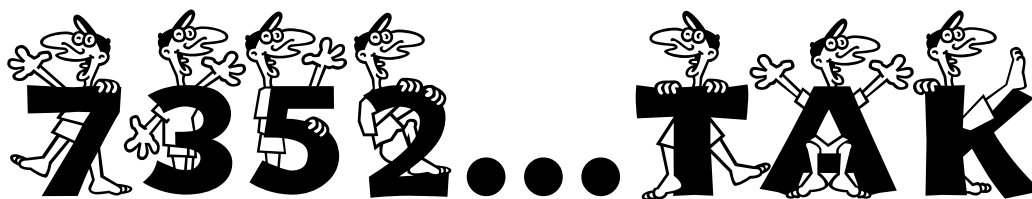
Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n , gdzie $0 \leq n \leq 10^9$

Wyjście: Na standardowe wyjście należy wypisać słowo TAK, jeśli podana liczba jest liczbą pierwszą, NIE – w przeciwnym wypadku.

2.4. SUMA CYFR

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n , gdzie $0 \leq n \leq 10^9$

Wyjście: Na standardowe wyjście należy wypisać słowo TAK, jeśli suma cyfr liczby n jest liczbą pierwszą, NIE – w przeciwnym wypadku.



2.5. Napisz program wypisujący wszystkie liczby pierwsze z podanego pliku.

Wejście: W pliku **zad2-5.txt** znajduje się 500 liczb całkowitych n , gdzie $1 \leq n \leq 10^6$, po jednej w każdym wierszu.

Wyjście: Do pliku **odp2-5.txt** należy wypisać wszystkie liczby pierwsze z pliku **zad2-5.txt**

2.6. Napisz program wypisujący z podanego pliku wszystkie liczby, które są kwadratami liczb pierwszych.

Wejście: W pliku **zad2-6.txt** znajduje się 500 liczb całkowitych n , gdzie $1 \leq n \leq 10^6$, po jednej w każdym wierszu.

Wyjście: Do pliku **odp2-6.txt** należy wypisać wszystkie liczby, które są kwadratami liczb pierwszych.

2.7. Napisz program wypisujący wszystkie liczby pierwsze z podanego przedziału (a, b) lub informację, że takich liczb w podanym przedziale nie ma.

Wejście: W pierwszym wierszu standardowego wejścia znajdują się dwie liczby całkowite a i b , gdzie $1 \leq a < b \leq 1000$

Wyjście: Na standardowe wyjście należy wypisać wszystkie liczby pierwsze z przedziału (a, b) lub komunikat: BRAK

Przykłady:

Dla danych wejściowych: 11 20 poprawną odpowiedzią jest: 13 17 19

Dla danych wejściowych: 19 23 poprawną odpowiedzią jest: BRAK

2.8. DZIELNIKI LICZB

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n , gdzie $1 \leq n \leq 10^9$

Wyjście: Na standardowe wyjście należy wypisać wszystkie dzielniki liczby n .

Przykład:

Dla danej wejściowej: 12

Poprawną odpowiedzią jest: 1 2 3 4 6 12

2.9. Napisz program wypisujący sumę wszystkich dzielników podanej liczby naturalnej.

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n , gdzie $1 \leq n \leq 10^9$

Wyjście: Na standardowe wyjście należy wypisać sumę wszystkich dzielników liczby n .

Przykład:

Dla danej wejściowej: 12

Poprawną odpowiedzią jest: 28

2.10. Napisz program wypisujący liczby mające dokładnie 3 dzielniki.

Wyjście: Na standardowe wyjście należy wypisać po spacji wszystkie liczby naturalne mniejsze od 500, które mają dokładnie 3 różne dzielniki.

2.11. ROZKŁAD LICZBY NA CZYNNIKI PIERWSZE

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n , gdzie $1 \leq n \leq 10^9$

Wyjście: Na standardowe wyjście należy wypisać listę czynników pierwszych danej liczby.

Przykład:

Dla danej wejściowej: 12

Wynikiem jest lista: 2 2 3 3

2.12. Napisz program wypisujący, ile liczba n ma różnych dzielników, które są liczbami pierwszymi.

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n , gdzie $1 \leq n \leq 10^9$

Wyjście: Na standardowe wyjście należy wypisać jedną liczbę, ilość różnych dzielników pierwszych liczby n .

Przykład:

Dla danej wejściowej: 12

Poprawnym wynikiem jest: 2

2.13. LICZBY DOSKONAŁE

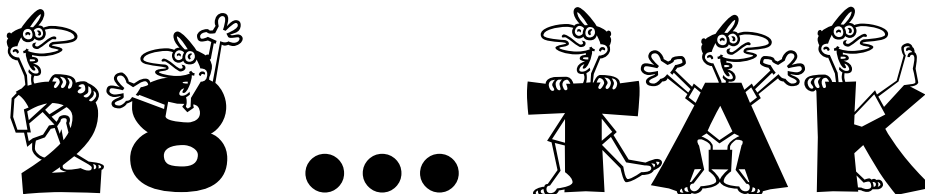
Liczbę nazywamy **doskonałą**, jeżeli jest sumą swoich dzielników właściwych tj. mniejszych od niej samej.

Na przykład liczba 6 jest liczbą doskonałą, ponieważ $6 = 1 + 2 + 3$

Napisz program sprawdzający, czy dana liczba jest doskonała.

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n , gdzie $1 \leq n \leq 10^9$

Wyjście: Na standardowe wyjście należy wypisać słowo TAK, jeśli podana liczba jest liczbą doskonałą, NIE – w przeciwnym wypadku.



2.14. Napisz program wypisujący do pliku liczby doskonałe.

Wyjście: Do pliku **odp2-14.txt** wypisz wszystkie liczby doskonałe mniejsze od miliona.

2.15. LICZBY DOSKONAŁE II RZĘDU.

Liczbą **doskonałą II rzędu** nazywamy liczbę naturalną $n > 1$, która jest równa iloczynowi wszystkich swoich dzielników mniejszych od niej samej.

Na przykład liczba 6 jest liczbą doskonałą II rzędu, ponieważ $6 = 1 * 2 * 3$

Wyjście: Do pliku **odp2-15.txt** wypisz wszystkie liczby doskonałe II rzędu mniejsze od 1000.

Na ekranie wypisz ile jest takich liczb.

2.16. LICZBY ZAPRZYJAŻNIONE

Liczby A i B nazywa się **zaprzyjażnionymi**, jeżeli suma wszystkich dzielników liczby A mniejszych od niej samej (sumaA) jest równa liczbie B i suma wszystkich dzielników liczby B mniejszych od niej samej (sumaB) jest równy liczbie A.

Pierwsza para takich liczb to 220 i 284.

Do pliku **odp2-16.txt** wypisz wszystkie pary liczb zaprzyjażnionych, z których co najmniej jedna jest mniejsza od miliona. Na ekranie wypisz ile jest takich par.

2.17. Napisz program wyszukujący w pliku pary liczb zaprzyjażnionych.

Wejście: W pliku **zad2-17.txt** znajduje się 1000 liczb naturalnych, po dwie w każdym wierszu. Każda z tych liczb jest mniejsza od 10^9 .

Wyjście: Do pliku **odp2-17.txt** wypisz wszystkie pary liczb zaprzyjażnionych. Na ekranie wypisz ile jest takich par.

2.18. Na podstawie numeru PESEL podaj liczbę kobiet w bazie danych.

Numer PESEL to 11-cyfrowy kod jednoznacznie identyfikujący określoną osobę fizyczną.

Dla wszystkich urodzonych w latach 1900-1999 skonstruowany został w następujący sposób:

- cyfry na pozycjach od 1 do 6 to data urodzenia (w kolejności: ostatnie dwie cyfry roku, numer miesiąca w postaci dwóch cyfr oraz numer dnia w postaci dwóch cyfr),
- cyfry na pozycjach od 7 do 9 tworzą liczbę porządkową,
- cyfra na pozycji 10 oznacza płeć (cyfra parzysta dla kobiet, nieparzysta dla mężczyzn),
- cyfra z pozycji 11 to cyfra kontrolna.

Dla osób urodzonych w roku 2000 oraz w późniejszych latach do numeru miesiąca dodana jest liczba 20. W ten sposób w numerze PESEL odróżniane są od siebie kolejne stulecia.

Wejście: W pliku **zad2-18.txt** znajdują się 400 wierszy z danymi osób zarejestrowanymi w systemie ewidencji. Każdy z wierszy składa się z następujących informacji: numer PESEL, nazwisko oraz imię. Poszczególne dane w wierszu rozdzielone są pojedynczą spacją.

Wyjście: Liczba całkowita oznaczająca liczbę kobiet, których pesel zapisany jest w pliku **zad2-18.txt**.

2.19. Sprawdzamy poprawność numeru PESEL

Cyfra kontrolna jest obliczana wg następującego algorytmu:

1. Każdą cyfrę numeru PESEL (oprócz ostatniej, która jest cyfrą kontrolną) mnożymy przez odpowiadającą jej wagę. Tablica wag ma postać: 1, 3, 7, 9, 1, 3, 7, 9, 1, 3.
2. Wyniki sumujemy.
3. Wyznaczamy resztę z dzielenia sumy przez 10
Jeżeli reszta = 0, to cyfra kontrolna wynosi 0.
Jeżeli reszta \neq 0, to cyfra kontrolna będzie uzupełnieniem reszty do 10,

Przykład dla numeru PESEL 44051401358:

$$1*4 + 3*4 + 7*0 + 9*5 + 1*1 + 3*4 + 7*0 + 9*1 + 1*3 + 3*5 = 101$$

Wyznaczamy resztę z dzielenia sumy przez 10: $101:10 = 10$ reszta = 1

Jeżeli reszta = 0, to cyfra kontrolna wynosi 0. Jeżeli reszta $\neq 0$, to cyfra kontrolna będzie uzupełnieniem reszty do 10, czyli w podanym przykładzie jest to cyfra 9.

$10 - 1 = 9$ Wynik 9 nie jest równy ostatniej cyfrze numeru PESEL, czyli 8, więc numer jest błędny.

Napisz program wyszukujący błędnie numery Pesel dla danych w pliku **zad2-19.txt**.

Wejście: W pliku **zad2-19.txt** znajdują się 400 wierszy z danymi osób zarejestrowanych w systemie ewidencji. Każdy z wierszy składa się z następujących informacji: numer PESEL, nazwisko oraz imię. Poszczególne dane w wierszu rozdzielone są pojedynczą spacją.

Wyjście: W pliku **odp2-19.txt** należy wypisać informacje (pesel, nazwisko i imię) wszystkich osób, których pesel jest niepoprawny.

2.20. ALGORYTM EUKLIDESA

Wejście: W pierwszym wierszu standardowego wejścia znajdują się dwie liczby całkowite a i b , gdzie $1 \leq a, b \leq 10^9$

Wyjście: Na standardowe wyjście należy wypisać największy wspólny dzielnik liczb a i b .

Przykład:

Dla danej wejściowej: 16 12

Poprawnym wynikiem jest: 4

2.21. Sprawdzenie, czy rozwinięcie dziesiętne ułamka jest skończone czy nieskończone.

Wejście: W pierwszym wierszu standardowego wejścia znajdują się dwie liczby całkowite l i m , gdzie $1 \leq l < m \leq 10^9$ oznaczające licznik i mianownik ułamka.

Wyjście: Na standardowe wyjście należy wypisać słowo „SKONCZONE” jeśli rozwinięcie dziesiętne ułamka $\frac{l}{m}$ jest skończone, „NIESKONCZONE – przeciwnym przypadku.

Przykłady:

Dla danych wejściowej: 6 8

Poprawnym wynikiem jest: SKONCZONE

Dla danych wejściowej: 2 6

Poprawnym wynikiem jest: NIESKONCZONE

2.22. CIĄG FIBONACCIEGO.

Ciąg Fibonacciego zdefiniowany jest następująco:

$$a_1 = 1, a_2 = 1, a_3 = 2, \dots, a_n = a_{n-1} + a_{n-2}$$

Oto kilka pierwszy wyrazów tego ciągu: 1 1 2 3 5 8 13 21 34 55 ...

Napisz program obliczający n -ty wyraz ciągu Fibonacciego, gdzie $1 \leq n \leq 50$.



2.23. SZYBKIE PODNOSZENIE DO POTĘGI

Dla danych dwóch liczb naturalnych a i n wyznacz wartość liczby a^n .

Wejście: Pierwszy i jedyny wiersz zawiera dwie liczby całkowite a i b , gdzie $1 \leq a, n \leq 100$

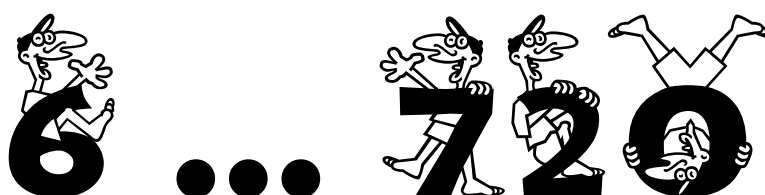
Wyjście: Program powinien wypisać ostatnią cyfrę działania a^n .

2.24. SILNIA

Silnia liczby naturalnej n – to iloczyn wszystkich liczb naturalnych nie większych niż n .

$n! = 1 * 2 * 3 * \dots * n$, oraz $0! = 1$

Oblicz wartość silni dla podanej liczby naturalnej $n < 40$



2.25. LICZBY DEFICYTOWE

Liczba deficytowa to taka liczba, której suma wszystkich dodatnich dzielników (za wyjątkiem jej samej) jest mniejsza od niej samej.

Przykłady liczb deficytowych: 1, 2, 3, 4, 5, 7, 8, 9, 10, 11

Napisz program wypisujący dwucyfrowe liczby deficytowe.

2.26. LICZBY PRONICZNE

Liczba proniczna to liczba będąca iloczynem dwóch następujących po sobie liczb naturalnych, tzn. $n(n + 1)$.

Przykłady liczb pronicznych: 2, 6, 12, 20, 30, 42, 56, 72, 90, 110

Np. $30 = 5 * 6$

Napisz program wypisujący wszystkie liczby proniczne mniejsze od 1000.

2.27. LICZBY CZWORACZE

Liczby czworacze to czwórki liczb pierwszych postaci p , $p+2$, $p+6$ i $p+8$ dające dwie pary liczb bliźniaczych. Napisz program, który wypisze kolejne liczby czworacze mniejsze od 100000.

Przykład liczb czworaczych:

5,7,11,13

11,13,17,19

101,103,107,109



2.28. LICZBY ARMSTRONGA (NARCYSTYCZNE)

Liczba Armstronga (narcystyczne) to - n-cyfrowa liczba naturalna, która jest sumą swoich cyfr podniesionych do potęgi n. Przykład - liczbą Armstronga jest $371 = 3^3 + 7^3 + 1^3$.
Napisz program, który znajdzie wszystkie czterocyfrowe liczby narcystyczne.

2.29. WESOŁE LICZBY

Liczba wesoła - liczba naturalna zdefiniowaną w następujący sposób:

- Obliczamy sumę kwadratów cyfr składających się na liczbę.
- Powtarzamy tę operację dla kolejnych wyników tak długo, aż uzyskamy liczbę 1 lub wyniki zaczną się powtarzać.

Jeżeli w wyniku procesu otrzymaliśmy 1, pierwotna liczba jest liczbą wesołą.

W przeciwnym przypadku jest liczbą smutną.

7 jest liczbą wesołą ponieważ podlega następującej sekwencji obliczeń:

$$7^2 = 49$$

$$4^2 + 9^2 = 97$$

$$9^2 + 7^2 = 130$$

$$1^2 + 3^2 + 0^2 = 10$$

$$1^2 + 0^2 = 1.$$

19 jest liczbą wesołą ponieważ podlega następującej sekwencji obliczeń:

$$1^2 + 9^2 = 82$$

$$8^2 + 2^2 = 68$$

$$6^2 + 8^2 = 100$$

$$1^2 + 0^2 + 0^2 = 1.$$

Napisz program zliczający ilość liczb wesołych mniejszych od 100.



2.30. LICZBY LUSTRZANE

Liczby lustrzane to takie dwie liczby, które są nawzajem swoim lustrzanym odbiciem np. 78 i 87, 197 i 791, 7682 i 2867

Napisz program sprawdzający, czy dwie podane liczby a i b są lustrzane.

Wejście: W pierwszym wierszu standardowego wejścia znajdują się dwie liczby naturalne a i b ($1 \leq a, b \leq 10^6$)

Wyjście: Na standardowe wyjście należy wypisać słowo TAK, jeśli podana liczba jest lustrzana, NIE – w przeciwnym wypadku.

2.31. LICZBY WAMPIRZE

Liczba wampirza n posiada parzystą liczbę cyfr. Musi być równa iloczynowi dwóch liczb utworzonych z cyfr danej liczby n. Cyfry mogą być wybierane z liczby n w dowolnej kolejności

i muszą zawierać połowę cyfr liczby wyjściowej. Nie dopuszcza się w liczbie dwóch zer na końcu liczby.

Przykłady liczb wampirzych:

$$1260 = 21 * 60$$

$$1827 = 21 * 87$$

$$2187 = 27 * 81$$

Napisz program, który wyszuka wszystkie czterocyfrowe liczby wampirze.

2.32. LICZBY SFENICZNE

Liczby sfeniczne to liczby naturalne, które są iloczynem dokładnie trzech różnych liczb pierwszych. Wszystkie liczby sfeniczne mają dokładnie osiem dzielników.

Napisz program, który sprawdzi, czy dana liczba naturalna n ($1 \leq n \leq 500$) jest liczbą sfeniczną. Jeśli jest – program wypisuje odnalezione liczby pierwsze, których iloczyn jest równy liczbie n , w przeciwnym przypadku wysyła komunikat - NIE.

Przykłady liczb sfenicznych:

$$70 = 2 * 5 * 7$$

$$273 = 3 * 7 * 13$$

$$438 = 2 * 3 * 73$$

2.33. LICZBY PÓŁPIERWSZE

Liczbami półpierwszymi nazywamy liczby, które są iloczynem dwóch liczb pierwszych.

Przykłady liczb półpierwszych: 34, 95, 699, 841

$$\text{Uzasadnienie: } 34 = 2 * 17 \quad 95 = 5 * 19 \quad 699 = 3 * 233 \quad 841 = 29 * 29$$

W pliku **zad2-33.txt** umieszczono w kolejnych wierszach 500 całkowitych liczb dodatnich, z których każda ma co najwyżej 6 cyfr. Napisz program, za pomocą którego otrzymasz wszystkie liczby półpierwsze z pliku **zad2-33.txt**. Otrzymane wyniki zapisz w pliku **odp2-33.txt**.

2.34. LICZBY P_SUMOPODOBNE.

Dodatnia liczba całkowita n jest **P_sumopodobna**, jeśli suma jej cyfr w systemie dziesiętnym jest równa sumie jej cyfr w reprezentacji przy podstawie P , gdzie $2 \leq P < 10$.

Przykłady:

Liczba 21 jest 2_sumopodobna, bo $21 = (10101)_2$ i $2 + 1 = 1 + 0 + 1 + 0 + 1 = 3$

Liczba 23 jest 3_sumopodobna, bo $23 = (212)_3$ i $2 + 3 = 2 + 1 + 2 = 5$

Napisz program sprawdzający, czy podana liczba n jest P_sumopodobna. Jeśli TAK – należy wypisać wszystkie P , dla których warunek zachodzi, lub wypisać NIE – jeśli liczba n nie jest P_sumopodobna.

2.35. Obliczanie wartości $\sin(x)$, $\cos(x)$.

Istnieją wzory na obliczenie za pomocą komputera w sposób przybliżony wartości funkcji sinus i cosinus.

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots$$

$$\cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \dots$$

Napisz program, który wyliczy wartości funkcji sinus i cosinus dla podanego przez użytkownika argumentu x (typu double) z dokładnością do 0,0001. Porównaj otrzymane wyniki z wartościami wskazanymi za pomocą funkcji $\sin(x)$ oraz $\cos(x)$ z biblioteki `<cmath>`.

2.36. SORTOWANIE PUNKTÓW

W pliku **zad2.36.txt** znajduje się 50 par liczb całkowitych (x, y) z przedziału $<-100; 100>$. Każde dwie liczby x, y opisują inny punkt na płaszczyźnie. Posortuj te punkty niemalejąco względem pierwszej współrzędnej, a jeśli te współrzędne są równe, względem drugiej współrzędnej.

2.37. WYDAWANIE RESZTY

Mając daną kwotę K i zestaw dostępnych nominałów banknotów i monet. Napisz program, który znajdzie najmniejszą liczbę banknotów i monet potrzebną do wydania kwoty równej K lub wypisze informację, że kwoty K nie da się wypłacić.

Wejście:

K - liczba naturalna oznaczająca kwotę do wydania

N - liczba naturalna oznaczająca liczbę dostępnych nominałów

$tab[N] = \{n_1, n_2, \dots, n_N\}$ – posortowana malejąco tablica dostępnych wartości nominałów

Wynik:

Liczba naturalna równa najmniejszej liczbie banknotów i monet, za pomocą których można wydać kwotę równą K lub informacja, że nie można wypłacić żądanej kwoty

Przykład 1

Dla danych wejściowych:

53

7

100 50 20 10 5 2 1

Poprawną odpowiedzią jest: 3

Przykład 2

Dla danych wejściowych:

53

4

20 10 5 1

Poprawną odpowiedzią jest: 6

Przykład 2

Dla danych wejściowych:

53

5

100 20 10 5 2

Poprawną odpowiedzią jest: NIE

3.OSIEM MIESIĘCY PÓŹNIEJ ... ZADANIA MATURALNE

Jeżeli zrobiłeś/łaś zadania z poprzednich rozdziałów i **przedyskutowałeś/łaś swoje rozwiązania z nauczycielem** już czas zmierzyć się z wybranymi zadaniami maturalnymi z informatyki.

Klucze rozwiązań znajdują się na stronie Centralnej Komisji Egzaminacyjnej w Warszawie oraz stronach Okręgowych Komisji Egzaminacyjnych.



3.1. LICZBY (maj 2016)

W pliku **zad3-1.txt** znajduje się 2000 liczb całkowitych z zakresu $< 0; 30000$ >

3.1.1. Podaj, ile liczb zapisanych w pliku **zad3-1.txt** to liczby pierwsze

3.1.2. Podaj, jaka jest największa i najmniejsza liczba w pliku **zad3-1.txt**

3.1.3. Liczby bliźniacze to takie dwie liczby pierwsze, które różnią się o 2,
np. (3, 5), (5, 3), (11, 13), czy (19, 17), itd.

Które sąsiadujące ze sobą liczby z pliku to liczby bliźniacze, tzn. badamy pierwszą liczbę z drugą, drugą z trzecią, ..., przedostatnią z ostatnią. Wypisz wszystkie takie pary do pliku **odp3-1.txt**, każdą parę w osobnym wierszu, oraz wypisz na ekranie liczbę znalezionych par liczb bliźniaczych.

3.2. LICZBY SKOJARZONE (maj 2016)

Dwie różne liczby całkowite A i B większe od 1 nazywamy skojarzonymi, jeśli suma wszystkich różnych dodatnich dzielników A mniejszych od A jest równa $B + 1$, a suma wszystkich różnych dodatnich dzielników B mniejszych od B jest równa $A + 1$.

Napisz program, który dla podanej liczby całkowitej $A > 1$ znajdzie i wypisze liczbę B skojarzoną A lub komunikat *NIE*, jeśli taka liczba nie istnieje.

3.3. NAPISY (maj 2013)

W pliku **zad3-3.txt** znajduje się 1000 napisów o długościach od 2 do 16 znaków, każdy napis w osobnym wierszu. W każdym napisie mogą wystąpić jedynie dwa znaki: „0” lub „1”.

a) Podaj, ile jest napisów o parzystej długości.

b) Podaj, ile jest napisów, które zawierają taką samą liczbę zer i jedynek.

c) Podaj, ile jest napisów składających się z samych zer, oraz podaj, ile jest napisów składających się z samych jedynek.

d) Dla każdej liczby $k = 2, 3, \dots, 16$ podaj liczbę napisów o długości k znajdujących się w pliku **napisy.txt**, tzn. podaj, ile jest napisów 2-znakowych, ile jest napisów 3-znakowych itd.

3.4. SZYFR CEZARA (maj 2016). Napisz program, który zaszyfruje teksty szyfrem Cezara z kluczem $k = 107$

W pliku **zad3-4.txt** znajduje się 100 słów, każde w osobnym wierszu. Zaszyfruj wszystkie słowa szyfrem Cezara z kluczem $k = 107$. Wszystkie wyniki zapisz w pliku **odp3-4.txt**.

Przykład. Dla słowa: INTERPRETOWANIE

poprawnym wynikiem jest: LQWHUSUHWZRZDQLH

3.5. SZYFR CEZARA (maj 2016). Napisz program, który odszyfruje teksty zaszyfrowane szyfrem Cezara.

W pliku **zad3-5.txt** znajduje się 500 szyfrogramów i odpowiadające im klucze szyfrujące. W każdym wierszu znajduje się jeden szyfrogram (zaszyfrowane słowo) i po pojedynczym znaku odstępu odpowiadający mu klucz (maksymalnie czterocyfrowa liczba).

Fragment pliku **zad3-5.txt**:

BCYKUNCM 1718

WARDA 9334

Napisz program, który odszyfruje słowa zaszyfrowane podanymi kluczami. Wynik zapisz w pliku **odp3-5.txt**, każde odszyfrowane słowo w osobnym wierszu.

3.6. SZYFR CEZARA (maj 2016). Napisz program, który sprawdzi, które szyfrogramy są błędnie zaszyfrowane metodą Cezara.

W pliku **zad3-6.txt** znajduje się 3000 par słów, po jednej parze w wierszu, oddzielone pojedynczym znakiem odstępu. Drugie słowo w każdej parze jest szyfrogramem pierwszego z nieznanym kluczem.

Niektóre szyfrogramy są błędne, co oznacza, że niektóre litery w słowie zakodowano z różnymi przesunięciami. Słowo ma zawsze tę samą długość co odpowiadający mu szyfrogram.

Fragment pliku **zad3-6.txt**:

ZAWISLAK EFBNXQFP

KRASZEWSKI XENFMRJFXV

Napisz program, który wyszuka i wypisze te słowa z pliku **zad3-6.txt**, które błędnie zaszyfrowano. Wynik wypisz w pliku **odp3-6.txt**, każde słowo w osobnym wierszu.

3.7. HASŁA (maj 2011). Napisz program/programy wykonujące poniższe polecenia:

W pliku **zad3-7.txt** znajduje się 200 haseł. Każde hasło składa się tylko z małych liter alfabetu angielskiego, zaś jego długość wynosi od 3 do 10 znaków.

Wyniki wypisz odpowiednio do plików **odp3-7a.txt**, **odp3-7b.txt**, **odp3-7c.txt**.

a) Podaj ile haseł ma parzystą, a ile nieparzystą liczbę znaków.

b) Utwórz zestawienie haseł (po jednym w każdym wierszu) które są palindromami.

- c) Zapisz zestawienie haseł zawierających w sobie dwa kolejne znaki, których suma kodów ASCII wynosi 220.

3.8. LICZBY SUPER PIERWSZE (maj 2007).

Liczba „super pierwsza” to taka liczba naturalna, która spełnia następujące warunki:

- jest liczbą pierwszą
- suma cyfr tej liczby jest również liczbą pierwszą

Liczba „super B pierwsza” oprócz wymienionych warunków spełnia dodatkowo

- Suma cyfr w jej zapisie binarnym jest także liczbą pierwszą

Napisz program zliczający, ile jest liczb „super B pierwszych” w przedziale

$< 100; 10000 >$

3.9. **PARY LICZB** (maj 2014). W pliku **zad3-9.txt** znajduje się 1000 par liczb. Każda para w osobnym wierszu. Liczby w parze oddzielone są spacją. Wszystkie liczby są całkowite dodatnie, nie większe niż 30000.

Napisz programy realizujące poniższe polecenia:

- a) Ile jest wierszy, w których jedna z występujących tam liczb jest wielokrotnością tej drugiej?
- b) Ile jest wierszy zawierających pary liczb względnie pierwszych, czyli takich, których największy wspólny dzielnik tych liczb jest równy 1?
- c) Ile jest wierszy, dla których suma cyfr pierwszej liczby jest równa sumie cyfr drugiej liczby?

3.10. **LICZBY FIBONACCIEGO** (styczeń 2013). W kolejnych wierszach pliku **zad3-10.txt** znajduje się 200 liczb naturalnych mniejszych od miliarda, po jednej w każdym wierszu. Napisz programy realizujące poniższe polecenia:

- a) Znajdź wszystkie liczby w pliku **zad3-10.txt**, które są liczbami Fibonacciego. Wynik zapisz w pliku **odp3-10.txt**.
- b) Wśród znalezionych liczb Fibonacciego, znajdź najmniejszą i największą
- c) Wśród znalezionych liczb Fibonacciego znajdź najdłuższy rosnący ciąg elementów. Jako odpowiedź podaj długość tego ciągu.

3.11. LICZBY BINARNE (maj 2015).

W pliku **zad3-11.txt** znajduje się 1000 liczb naturalnych zapisanych binarnie. Każda liczba zapisana jest w osobnym wierszu. Każda liczba binarna zawiera co najwyżej 250 cyfr binarnych. Pierwsze pięć wierszy zawiera następujące liczby:

11010100111

11110111111011101

1010100111010100

3.14. ANAGRAMY CYFROWE (przykładowy arkusz maturalny CKE)

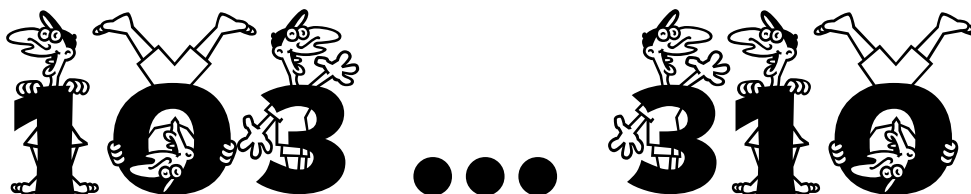
Powiemy, że dwie liczby naturalne a i b są **anagramami cyfrowymi**, jeśli liczbę a można zapisać dziesiętnie za pomocą cyfr występujących w zapisie liczby b używając każdej cyfry dokładnie tyle razy, ile razy występuje w zapisie b .

Uwaga: przyjmujemy, że 0 występuje na najbardziej znaczącej pozycji tylko w zapisie liczby zero.

W pliku **zad3-14.txt** znajduje się 1000 par liczb całkowitych dodatnich, z których każda jest nie większa niż 2000000. Każda para liczb jest zapisana w osobnym wierszu. Liczby w wierszu są oddzielone pojedynczym znakiem odstępu.

Napisz program(-y), który poda odpowiedzi na następujące pytania:

- W ilu wierszach pliku **zad3-14.txt** zapisane w nich liczby to anagramy cyfrowe?
- Ile maksymalnie liczb można znaleźć w pliku **zad3-14.txt**, z których każde dwie to anagramy cyfrowe?



3.15. SYSTEMY LICZBOWE (czerwiec 2016)

W pliku **zad3-15.txt** zapisano 999 kodów liczb, każdy kod w osobnym wierszu. Ostatnia cyfra kodu oznacza podstawę systemu liczbowego (od dwójkowego do dziewiątkowego), w jakim zapisano liczbę kolejnymi cyframi kodu, od pierwszej do przedostatniej.

Na przykład:

12345678 oznacza liczbę $1234567_{(8)}$, czyli liczbę 1234567 zapisaną w systemie ósemkowym. Każdy kod liczby zaczyna się cyfrą większą od 0, a jego długość (wraz z cyfrą oznaczającą system liczbowy) nie przekracza 10 cyfr.

Napisz programy, których wynikiem działania będą rozwiązania poniższych zadań. Wyniki zapisz w oddzielnych plikach.

- Podaj, ile liczb w pliku **zad3-15.txt** zapisano w systemie ósemkowym.
- Podaj, ile wierszy w pliku **zad3-15.txt** zawiera liczby zapisane w systemie czwórkowym takie, że w ich zapisie nie występuje cyfra 0.
- Podaj, ile wierszy w pliku **zad3-15.txt** zawiera liczby parzyste zapisane w systemie dwójkowym.
- Podaj sumę wszystkich liczb z **zad3-15.txt**, które zapisano w systemie ósemkowym. Wynik podaj w systemie dziesiętnym.
- Podaj kod największej oraz kod najmniejszej spośród liczb zakodowanych w pliku **zad3-15.txt** oraz ich wartości w systemie dziesiętnym.

4. JESZCZE W TYM ROKU ... ZASTOSOWANIA

4.1. Zbliża się ważna szkolna uroczystość „Święto Szkoły”. Na sali gimnastycznej ustawiono dla uczestników imprezy 500 krzesełek (5 rzędów po 100 krzesełek w rzędzie). Część osób już usiadło w wyznaczonych miejscach. Zadaniem ucznia Bartosza jest podanie informacji, w którym rzędzie jest najwięcej wolnych miejsc.

Wejście: W pliku **zad4-1.txt** znajduje się 500 cyfr (1 lub 0) w pięciu kolejnych wierszach, po 100 w każdym wierszu. Cyfra 1 oznacza zajęte miejsce, 0 – wolne krzesło.

Wyjście: Na standardowe wyjście należy wypisać jedną liczbę ze zbioru {1,2,3,4,5} oznaczającą numer rzędu w którym jest najwięcej wolnych miejsc.



4.2. Uroczystość Święta Szkoły uświetni występ szkolnego chóru. Chórzyści stoją w jednym, długim rzędzie w kształcie półkola. Część występujących uczniów jeszcze nie przyszła, a już czas zacząć próbę. Wykonawcy potrzebują nut, a Bartek musi się zatroszczyć o to, by osoba nie mająca nut stała bezpośrednio obok osoby z nutami.

Napisz program, który wyznaczy minimalną liczbę potrzebnych nut.

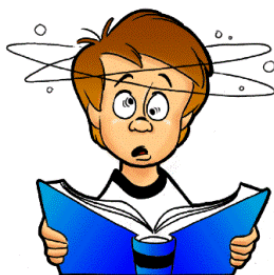
Wejście: W pierwszym wierszu standardowego wejścia znajduje się ciąg cyfr 1 i 0, gdzie 1 oznacza zajęte miejsce, 0 – wolne miejsce (uczeń jeszcze nie przyszedł). Łączna długość ciągu nie przekracza 1000 cyfr.

Wyjście: Na standardowe wyjście program powinien wypisać jedną liczbę całkowitą, minimalną liczbę nut, które Bartek musi rozdać.

Przykład:

Dla danych wejściowych: 011011101

Poprawną odpowiedzią jest: 3



4.3. Bartek chce wybrać z grona znajomych koleżanek, tę jedną, której zaproponuje wspólne wyjście do kina. Ponieważ wszystkie jego koleżanki są śliczne ciężko mu podjąć ostateczną decyzję. Postanowił zrobić test i powiedział, że film trwa ... i tu podał czas w sekundach. Pomóż dziewczętom zapisać wynik w formacie godziny:minuty:sekundy

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n , gdzie $1 \leq n \leq 10^6$ oznaczającą czas w sekundach

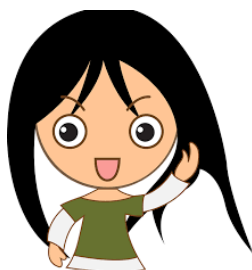
Wyjście: Na standardowe wyjście należy wypisać wynik w postaci: gg:mm:ss

Przykład

Dla danych wejściowych:

4000

Poprawnym wynikiem jest: 01:06:40



4.4. Bartek już wie, z którą dziewczyną umówi się do kina. Poprosił ją o numer telefonu. Aby łatwiej było go zapamiętać zamienił każdą cyfrę na literę zgodnie z pewną tabelą. Pomóż Bartkowi odszyfrować tak zapamiętany tekst.

cyfra	litera
9	ABC
8	DE
7	FGH
6	IJ
5	KLM
4	NO
3	PQR
2	ST
1	UVW
0	XYZ

Wejście: W pierwszym wierszu standardowego wejścia znajduje się ciąg wielkich liter alfabetu łacińskiego- tekstowa postać numeru telefonu.

Wyjście: Na standardowym wyjściu powinna pojawić się cyfrowa postać numeru telefonu.

Przykład

Dla danych wejściowych: ALAMAKOTA

Poprawnym wynikiem jest: 959595429



4.5. Bartek z koleżanką po seansie filmowym postanowili kontynuować spotkanie w większej grupie znajomych. Tam w ramach zabaw intelektualnych grano w grę o nazwie BUM. Napisz program wypisujący poprawny ciąg liczb i tekstów w grze BUM.

Gra BUM polega na tym, że gracze stają w kółku i jeden po drugim mówią kolejne liczby naturalne. Gdy liczba, którą gracz ma właśnie powiedzieć dzieli się przez 3 lub zawiera w swoim zapisie cyfrę 3, mówi zamiast tej liczby słowo BUM. Następny gracz kontynuuje zabawę od kolejnej liczby. Jeżeli gracz się pomyli tj. powie BUM, kiedy nie powinien lub nie powie BUM, kiedy powinien odpada. Reszta graczy kontynuuje grę od liczby, na której odpadł ten zawodnik.

Przed każdą rozgrywką gracze ustalają liczbę a , od której zaczynają grę oraz liczbę b , na której kończą grę. Wszyscy, którzy nie odpadną aż do liczby b zostają zwycięzcami.

Wejście: W pierwszym wierszu standardowego wejścia znajdują się dwie liczby całkowite nieujemne $1 \leq a, b \leq 10000$.

Wyjście: Program powinien wypisać po spacji wszystkie odpowiedzi zgodne z zasadami gry od a do b (włącznie). Każda odzywka powinna być liczbą lub słowem BUM.

Przykład:

Dla danych wejściowych

4 20

Poprawną odpowiedzią jest:

4 5 BUM 7 8 BUM 10 11 BUM BUM 14 BUM 16 17 BUM 19 20



4.6. Bartek chce zrobić na koleżance dobre wrażenie. Wymyślił, że wyda się dziewczynie bardziej elokwentny, gdy będzie wyrażał swoje opinie w zdaniach o jak największej liczbie odstępów, np. „Hej, czy jest OK?” jest lepsze niż „Pięknie dziś wyglądasz”.

Napisz program, który pozwoli szybko wybrać najlepsze według Bartka zdanie. Możesz założyć, że rozwiązanie zadania jest zawsze jednoznaczne i że między wyrazami jest dokładnie jedna spacja.

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n , ilość zdań do sprawdzenia. W kolejnych n liniach zapisane są kolejne zdania.

Wyjście: Program powinien wypisać najlepsze według kryterium Bartka zdanie.



4.7. Bartoszowi coraz bardziej zależy na bliższej znajomości z koleżanką. Dlatego poprosił ją o kolejny termin spotkania. Dziewczyna chętnie się zgodziła zapowiadając, że będzie w umówionym miejscu za n sekund. Bartek musi się jednak trochę postarać, bo liczba n będzie podana w systemie binarnym.

Napisz program, który pozwoli szybko zamienić liczbę n na czas w minutach.

Wejście: W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n , złożona wyłącznie z cyfr 1 lub 0 oznaczająca liczbę sekund zapisaną w reprezentacji binarnej. Ilość cyfr liczby n nie przekracza 16.

Wyjście: Program powinien wypisać jedną liczbę całkowitą, czas w pełnych minutach.

Przykład

Dla danych wejściowych: 1111110100100

Poprawnym wynikiem jest: 135



4.8. Mama koleżanki bardzo chce poznać Bartka, ponieważ zauważyła, że jej córka spędza z nim coraz więcej czasu. Postanowiła zaprosić go na rodzinny obiad. Napisała na kartce pewne słowo, a następnie zamieniła w nim niektóre litery na cyfry. W ten sposób ukryła wewnątrz słowa pewien ciąg liczb. Zadaniem Bartka jest wyznaczenie sumy tych ukrytych liczb. Wynik jest numerem domu w którym mieszka koleżanka wraz z rodziną.

Wejście: W pierwszym wierszu wejścia znajduje się napis składający się z n znaków, bez odstępów ($1 \leq n \leq 5000000000$). Każdy znak to litera lub cyfra.

Wyjście: Program powinien wypisać dokładnie jedną liczbę całkowitą równą sumie ukrytych w napisie liczb.

Przykład - Dla danych wejściowych: Alama16kotoworaczterypsi9myszy

Poprawną odpowiedzią jest: 25



4.9. Bartek walczy z nadwagą, ale ćwiczenia i dieta to nie są jego ulubione tematy. Postanowił zapisywać codziennie swoją wagę i pochwalić się koleżance, że robi postępy w odchudzaniu. Jednak waga nie spada równomiernie, a czasami, ku przerażeniu Bartka nawet wzrasta. Znajdź taki okres, w którym schudł najwięcej.

Wejście: W pierwszym wierszu wejścia znajduje się jedna liczba całkowita n , gdzie $2 \leq n \leq 10^9$ ilość zapisanych pomiarów.

W drugim wierszu wejścia jest n liczb rzeczywistych w_1, w_2, \dots, w_n , gdzie w_i oznacza wagę Bartka w i -tym pomiarze z dokładnością do 2 miejsc po przecinku, $20 \leq w_i \leq 300$.

Wyjście: Pierwszy i jedyny wiersz wyjścia powinien zawierać jedną liczbę rzeczywistą oznaczającą maksymalny ciągły spadek wagi Bartka.

Przykład. Dla danych wejściowych:

5

67.20 88.35 88.00 85.99 87.40

Poprawną odpowiedzią jest: 2.36



4.10. W szkole odbywają się głosowania w ramach konkursu „Moja książka”. To turniej literacki, w którym autorzy współzawodniczą ze sobą w tworzeniu własnych publikacji. Zadaniem Bartka jest wypisanie imienia osoby, która uzyskała największą liczbę głosów, a w przypadku remisu – imion wszystkich osób z maksymalną punktacją.

Wejście: W pierwszej linii wejścia znajduje się jedna liczba n ($1 \leq n \leq 50$) oznaczająca liczbę uczestników. W każdym z następnych n wierszy znajduje się imię uczestnika oraz po spacji liczba oddanych na niego głosów.

Wyjście: Program powinien w kolejnych liniach wypisać imiona osób z największą liczbą głosów.

Przykład: Dla danych wejściowych

4

Dorota 125

Kasia 126

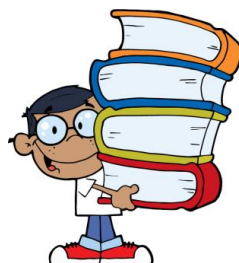
Nikola 118

Ewa 126

Poprawną odpowiedzią jest:

Kasia

Ewa



4.11. Plik z wynikami konkursu musi być utajniony, aż do uroczystości Święta Szkoły. Bartek zapisując dokument zaszyfrował go dla bezpieczeństwa pewnym hasłem. Dla pewności hasło umieścił wśród innych tekstów w pliku **zad4-11.txt**.

Podaj liczbę haseł z pliku **zad4.11.txt**, które spełniają poniższe kryteria:

- hasło ma dokładnie 8 znaków
- hasło zawiera co najmniej jeden znak numeryczny, tzn. cyfrę od 0 do 9,
- hasło zawiera co najmniej jedną małą literę,
- hasło zawiera co najmniej jedną dużą literę.

Wejście: W pliku **zad4.11.txt** znajduje się 100 haseł zawierających od 1 do 20 znaków alfanumerycznych, tzn. cyfr od 0 do 9 lub małych lub dużych liter alfabetu łacińskiego.

Wyjście: Na standardowym wyjściu - jedna liczba całkowita spełniająca powyższe kryteria.



4.12. Bartek postanowił napisać list miłosny do koleżanki. Aby tekst nie wpadł w niepowołane ręce zaszyfrował go specjalnie wymyślonym przez siebie szyfrem BART, a koleżance przekazał program deszyfrujący.

BART to kodowanie zamieniające każdy znak ASCII z przedziału 32 – 126 na znak znajdujący się o 10 pozycji dalej, ale nie dalej niż do 126 pozycji.

Napisz program szyfrujący i deszyfrujący podaną metodą dowolny ciąg znaków.



4.13. W IV Liceum Ogólnokształcącym w Białymstoku realizowana jest innowacja pedagogiczna „Szachy edukacją przyszłości – szachista lepszym uczniem”. Bartek chciałby przećwiczyć ruchy poszczególnych figur. Rysunek pokazuje w jakich kierunkach porusza się goniec.

Znając położenie dwóch wrogich sobie gońców Bartek powinien stwierdzić, czy obie figury się atakują. Dla uproszczenia literę na szachownicy zamienimy na liczby.

Wejście:

W dwóch wierszach wejścia podane są po dwie liczby całkowite dodatnie w i k będące odpowiednio numerem wiersza i kolumny pola, na którym stoją gońce ($1 \leq w, k \leq 8$).



Wyjście

Na standardowym wyjściu powinna znaleźć się odpowiedź TAK - jeśli te pionki atakują się, NIE- w przeciwnym wypadku.

Przykład 1

Dla danych wejściowych:

4 4

3 1

poprawną odpowiedzią jest:

NIE

Przykład 2

Dla danych wejściowych:

2 7

7 2

poprawną odpowiedzią jest:

TAK

4.14. Dzięki szachom Bartek, uczeń klasy informatycznej ma tak wytrenowany mózg, że potrafi przewidzieć konsekwencje nieodrobionej pracy domowej do sześciu pokoleń włącznie. Teraz ustawiając Królową (czasami zwaną Hetmanem) chciałby wiedzieć ile pól ona atakuje.

Wejście: W pierwszej linii wejścia znajdują się dwie liczby w oraz k , odpowiednio numer wiersza i kolumny pola, na którym stoi Królowa ($1 \leq w, k \leq 8$).

Wyjście

W pierwszym i jedynym wierszu powinna znaleźć się liczba naturalna - ilość pól jaką atakuje Królowa stojąca na podanym miejscu.



Przykład

Dla danych wejściowych:

4 4

poprawną odpowiedzią jest:

27

4.15. Mając podaną pozycję Króla oraz wybranego punktu docelowego Bartek musi podać w jakiej minimalnej liczbie ruchów Król może tam dotrzeć.

Wejście

W jednym wierszu wejścia podane są dwie liczby całkowite dodatnie w_1 i k_1 będące odpowiednio numerem wiersza i kolumny pola, z którego startuje Król. ($1 \leq w_1, k_1 \leq 8$). W kolejnym wierszu znajdują się dwie liczby całkowite w_2, k_2 ($1 \leq w_2, k_2 \leq 8$) oznaczające pozycję docelową Króla.



Wyjście

Liczba naturalna – informacja w ilu najmniej ruchach Król może przejść od pozycji startowej do pozycji docelowej.

Informacje o dopuszczalnych ruchach Króla znajdziesz z pewnością w Internecie.

Przykład

Dla danych wejściowych:

4 6

2 3

Poprawną odpowiedzią jest:

3

4.16. Mając podane pozycje dwóch skoczków Bartek musi stwierdzić, czy się one atakują.

Dopuszczalne ruchy skoczka oznaczone są na rysunku kółeczkami.

Wejście

W jednym wierszu wejścia podane są dwie liczby całkowite dodatnie w_1 i k_1 będące odpowiednio numerem wiersza i kolumny pola, na którym znajduje się pierwszy skoczek. W drugim wierszu wejścia podane są dwie liczby całkowite dodatnie w_2 i k_2 będące odpowiednio numerem wiersza i kolumny pola, drugiego skoczka.



Wyjście

Słowo TAK, jeśli skoczki się atakują, NIE w przeciwnym wypadku.



Przykład 1

Dla danych wejściowych:

4 5

3 7

Poprawną odpowiedzią jest TAK

Przykład 2

Dla danych wejściowych:

1 1

8 8

Poprawną odpowiedzią jest NIE

4.17. Niedługo wakacje. Bartek z przyjaciółmi zamierzają wziąć udział w spływie kajakowym. Jaką minimalną liczbę dwuosobowych kajaków powinni wypożyczyć?

Wejście: W pierwszej linii wejścia znajdują się dwie liczby n oraz t

($1 \leq n \leq 50$, $10 \leq t \leq 300$) oznaczającą odpowiednio liczbę uczestników i ładowność każdego z kajaków. W następnym wierszu znajduje się n liczb całkowitych dodatnich określających wagę (w) kolejnego uczestnika ($1 \leq w \leq 300$)

Wyjście: Program powinien wypisać jedną liczbę całkowitą, liczbę kajaków które muszą być wypożyczone.

4.18. Nad bezpieczeństwem kajakarzy czuwać ma ratownik, który płynie jako ostatni. Wzrok ma jednak coraz słabszy i osoby znajdujące się w pewnej odległości są dla niego niewidoczne. Zadaniem Bartka jest określenie, ile osób w danej chwili jest poza zasięgiem wzroku ratownika.

Wejście: W pierwszej linii wejścia znajdują się cztery liczby całkowite n, w, x, y ($1 \leq n \leq 50$, $1 \leq w \leq 1000$, $0 \leq x, y \leq 1000$) oznaczające kolejno: liczbę uczestników spływu kajakowego (n), zasięg wzroku ratownika (w) i jego położenie (x, y).

W kolejnych n wierszach znajdują się wartości (x_i, y_i) oznaczające współrzędne i -tego kajakarza.

Wyjście: W pierwszym wierszu standardowego wyjścia należy wypisać liczbę uczestników spływu kajakowego, które są poza zasięgiem wzroku ratownika.

Przykład

Dla danych wejściowych:

5 3 4 2

2 4

3 3

4 5

5 6

2 3

Poprawną odpowiedzią jest: 2





5. NA KOLEJNY ROK ...

5.1. PAKOWANIE PLECAKA, CZYLI ... ZACHŁANNIE i DYNAMICZNIE

Mamy do dyspozycji 6 różnych produktów o podanych wagach i cenach oraz plecak o pojemności 12 kg. Twoim zadaniem jest wybór produktów do plecaka tak, by ich łączna suma wartości była jak największa i jednocześnie mieściły się w plecaku, czyli suma wag nie przekroczyła 12 kg.

Waga (kg)	4	5	3	2	1	5
Wartość (zł)	6	7	5	4	3	9
Nr przedmiotu	1	2	3	4	5	6

5.2. WYSZUKIWANIE BINARNE, CZYLI ... ROZWIĄŻ RÓWNANIE $x^3 + px = q$

Wejście: W pierwszej linii wejścia znajdują się dwie liczby całkowite p i q , gdzie $0 \leq p \leq 10^{12}$, $0 \leq q \leq 10^{18}$ oddzielone pojedynczym odstępem.

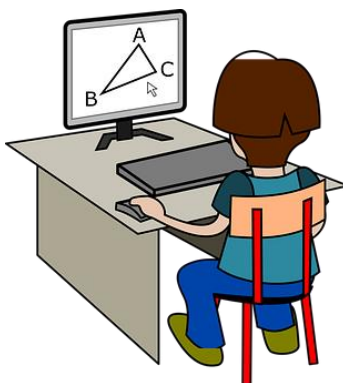
Wyjście: Liczba naturalna x , która spełnia warunki równania lub komentarz NIE, jeśli równanie nie ma rozwiązania w zbiorze liczb całkowitych.

Przykład 1: Dla danych wejściowych: 2 33

Poprawnym wynikiem jest: 3

Przykład 2: Dla danych wejściowych: 2 13

Poprawnym wynikiem jest: NIE



5.3. SUMOWANIE KOLEJNYCH LICZB PIERWSZYCH, CZYLI ... SUMY PREFIKSOWE

Napisz program obliczający sumę liczb pierwszych w podanym przedziale liczbowym.

Wejście: W pierwszej linii wejścia znajduje się jedna liczba naturalna $n < 100000$ oznaczającą ilość zestawów do sprawdzenia. W kolejnych n wierszach znajdują się dwie oddzielone spacją liczby całkowite p i q , gdzie $2 \leq p \leq q \leq 2000000$.

Wyjście: W kolejnych n wierszach podaj wynik sumy liczb pierwszych z podanego przedziału $< p, q >$

Przykład:

Dla danych wejściowych:

2

5 15

48 51

Poprawnym wynikiem jest:

36

0

5.4. ARYTMETYKA NA TEKSTACH, CZYLI ... DODAWANIE DUŻYCH LICZB

Napisz program obliczający sumę dwóch liczb a i b .

Wejście: W pierwszej linii standardowego wejścia znajdują się, oddzielone spacją liczby całkowite a i b , gdzie $0 \leq a \leq 10^{200}$, $0 \leq b \leq 10^{200}$

Wyjście: W pierwszej linii standardowego wyjścia powinna zostać wypisana jedna liczba całkowita, suma liczb podanych na wejściu.

5.5. ODWROTNA NOTACJA POLSKA (ONP), CZYLI ... WYRAŻENIA BEZNAWIASOWE

Odwrotną notację polską ONP wymyślono w celu zapisywania dowolnych wyrażeń arytmetycznych bez nawiasów. Na przykład wyrażenie: $(5 + 2) * 3$ można zapisać jako: $52 + 3 *$

Napisz program obliczający wartość wyrażenia zapisanego w postaci ONP. Załóż, że wyrażenie wejściowe jest zawsze poprawne i na wejściu nie ma liczb ujemnych.

Wejście: W pierwszej linii standardowego wejścia znajduje się wyrażenie w postaci ONP, gdzie każda liczba jest jednocyfrowa. Operatorami arytmetycznymi są 4 podstawowe działania: + dodawanie, - odejmowanie, * mnożenie, / dzielenie.

Ostatnim elementem wyrażenia jest znak "=". Powoduje on zakończenie obliczeń i wyprowadzenie wyniku.

Wyjście: Liczba całkowita, wynik obliczeń dla wyrażenia podanego na wejściu.

Przykład:

Dla danych wejściowych:

$52 + 3 * =$

Poprawnym wynikiem jest:

21

5.6. METODY SORTOWANIA, CZYLI ... PORZĄDKOWANIE DANYCH

Zakładamy, że mamy do dyspozycji dwa ciągi liczbowe, posortowane niemalejąco. Napisz program, który liczby z obu zbiorów wypisze w kolejności niemalejącej.

Wejście: W pierwszej linii standardowego wejścia znajdują się liczby n i m , oznaczające odpowiednio n - liczbę elementów pierwszego zbioru, m - liczbę elementów drugiego zbioru oraz $1 \leq n, m \leq 50$.



W drugim wierszu znajduje się n liczb oddzielonych spacją – elementy pierwszego zbioru.

W trzecim wierszu znajduje się m liczb oddzielonych spacją – elementy drugiego zbioru.

Wszystkie elementy obu zbiorów należą do przedziału $< -4 \cdot 10^9, 4 \cdot 10^9 >$.

Wyjście: Program powinien wypisać kolejno, po spacji $n + m$ liczb, elementy zbiorów podanych na wejściu, posortowane niemalejąco.

Przykład:

Dla danych wejściowych:

4 5

4 6 9 12

2 2 4 5 6

Poprawnym wynikiem jest:

2 2 4 4 5 6 6 9 12



RECENZJE

Zadania uporządkowane są według wzrastającego stopnia trudności, z sugerowanym czasem realizacji poszczególnych części. Interesujące jest wstawienie modułów „Jeszcze w tym roku ... zastosowania” , który zawiera zadania praktyczne oraz zadań rozwojowych „Na kolejny rok ...”. Opis zadań jest precyzyjny, zadania różnorodne.

Fragment recenzji Zdzisława Babicza

Zdecydowanym plusem publikacji jest różnorodność problemów i szczegółów algorytmicznych, które należy zauważyć w proponowanych ćwiczeniach: użycie właściwych dla specyfikacji typów danych, dobór poprawnej metody rozwiązania, czy efektywność algorytmów to tylko niektóre z walorów poznawczych poszczególnych zadań.

Pomysły na zadania nie są przypadkowe, cechuje je praktyczne podejście do zagadnienia, są bliskie uczniom, bo zaczerpnięte z ich życia codziennego. Treść wielu z nich zawiera pewną dozę żartu, zobrazowaną również humorystycznymi rysunkami.

Śmiało można polecić tę książkę wszystkim, którzy chcą pogłębić swoje umiejętności w zakresie algorytmiki i programowania.

Fragment recenzji Urszuli Ciłko



SPIS TREŚCI:

NA DOBRY POCZĄTEK ... KILKA WSKAZÓWEK	2
ZACZYNAMY ... OD PODSTAW (C++)	3
CZTERY MIESIĄCE PÓŹNIEJ ... ALGORYTMY	12
OSIEM MIESIĘCY PÓŹNIEJ ... ZADANIA MATURALNE	22
JESZCZE W TYM ROKU ... ZASTOSOWANIA	27
RECENZJE.....	39