

Zadania proste

1. Trójką pitagorejską nazywa się w matematyce trzy liczby naturalne a , b , c spełniające warunek $a^2 + b^2 = c^2$. Proszę napisać program wczytujący liczbę naturalną N i wypisujący wszystkie trójki w których c nie przekracza N .
2. Proszę napisać program który wczytuje wprowadzoną z klawiatury liczbę naturalną i odpowiada na pytanie, czy liczba ta jest wielokrotnością kwadratu dowolnej liczby naturalnej większej od 1 i nie jest równocześnie wielokrotnością sześciangu dowolnej liczby naturalnej większej od 1.
3. Liczbę $f(i)$ nazywamy liczbą Fibonacciego, gdy spełnia równania:
$$f(i) = 1 \quad \text{dla } i=1,2$$
$$f(i) = f(i-1) + f(i-2) \quad \text{dla } i>2$$

Proszę napisać program który wczytuje wprowadzoną z klawiatury liczbę naturalną i odpowiada na pytanie, czy liczba ta jest sumą co najwyżej czterech kolejnych liczb Fibonacciego.
4. Proszę napisać procedurę zamieniającą nieujemną liczbę całkowitą na system o podstawie p i wypisująca jej wartość w nowym systemie. Do procedury należy przekazać liczbę i i podstawę systemu, na który zamieniamy.
5. Proszę napisać procedurę wypisującą wartość ułamka A/B z dokładnością do N cyfr dziesiętnych. Do procedury należy przekazać: licznik, mianownik oraz liczbę określającą ilość miejsc po przecinku.
6. Dane są ciągi $a(n)$, $b(n)$ i $c(n)$ określone następująco:
$$a(n) = 1 \quad \text{dla } n=1, 2$$
$$a(n) = a(n-1) + a(n-2) \quad \text{dla } n>2$$
$$b(n) = 1 \quad \text{dla } n=1, 2, 3$$
$$b(n) = b(n-1) + b(n-2) + b(n-3) \quad \text{dla } n>3$$

Wyrazy ciągu $c(n)$ są kolejnymi liczbami naturalnymi należącymi do ciągu $a(n)$ lub $b(n)$. Ciągi te przyjmują wartości:
ciąg $a(n)$ 1 1 2 3 5 8 13 21 ...
ciąg $b(n)$ 1 1 1 3 5 9 17 31 ...
ciąg $c(n)$ 1 2 3 5 8 9 13 17 ...

Proszę napisać program który wczytuje wprowadzoną z klawiatury liczbę naturalną i i wypisuje kolejne wyrazy ciągu $c(n)$ mniejsze od wprowadzonej liczby.

Zadania z tablicami

1. Dane są definicje:

```
const
    max1 = 100;
    max2 = 2000;
type
    table = array[1..max1] of array[1..max2] of real;
```

Typ `table` opisuje tablicę zawierającą `max1` wierszy po `max2` elementów. Proszę napisać funkcję, która dla zmiennej typu `table` zwraca numer wiersza w którym największa wartość jest najmniejsza.

2. Dane są deklaracje:

```
const
    max1 = 100;
    max2 = 200;
type
    tablica = array[1..max1, 1..max2] of integer;
```

```
var
    T: tablica;
```

Zakładamy, że zmienna typu `tablica` jest wypełniona przypadkowymi wartościami. Proszę napisać procedurę, która dla zmiennej typu `tablica` zwraca współrzędne elementu `T[i, j]` dla którego iloraz sumy elementów w *i*-tej kolumnie i sumy elementów w *j*-tym wierszu jest określony (uwaga na dzielenie przez 0) i jest największy.

3. Dane są definicje:

```
const
    max1 = 100;
    max2 = 200;

type
    wiersz = array[1..max1] of integer;
    tablica = array[1..max2] of wiersz;
```

Zakładamy, że zmienna typu `tablica` jest wypełniona przypadkowymi wartościami. Proszę napisać funkcję, która dla zmiennej typu `tablica` zwraca numer wiersza w którym występuje najdłuższy spójny fragment złożony z liczb o tej samej wartości. W przypadku kilku fragmentów o tej samej długości należy zwrócić pozycję pierwszego z nich.

4. Dane są definicje:

```
const
    max = 100;
    bit = 0..1;

type
    longint = array[0..max] of bit;
```

Typ `longint` opisuje liczbę stałoprzecinkową kodowaną w kodzie U2. Bit znaku znajduje się na pozycji 0. Proszę napisać procedurę zmieniającą znak liczby typu `longint`. Procedura powinna sygnalizować wystąpienie nadmiaru.

5. Dane są deklaracje:

```
const
    max = 20;

type
    tablica = array[1..max, 1..max] of integer;

var
    T: tablica;
```

Proszę napisać funkcję, która zwraca wartość `true` jeżeli w zmiennej typu `tablica` istnieje przynajmniej jedna para sąsiednich elementów, których suma jest równa średniej arytmetycznej wszystkich elementów tablicy.

6. Dane są następujące definicje:

```
const
    max1 = 200;
    max2 = 20;

type
    d1 = array[1..max1] of boolean;
    d2 = array[1..max2] of d1;
```

Typ `d1` opisuje długą, dodatnią liczbę całkowitą. Wartość najbardziej znaczącego bitu znajduje się w elemencie o indeksie 1; wartość `true` oznacza wartość bitu 1. Proszę napisać procedurę porządkującą tablice opisana typem `d2` wg rosnących wartości liczb metodą przepisania ich do drugiej tablicy. Do procedury należy przekazać dwie tablice typu `d2`.

7. Dane są deklaracje:

```
const
    max = 20;

type
    tablica = array[1..max, 1..max] of integer;

var
    T: tablica;
```

Proszę napisać procedurę, która zmienną typu `tablica` wypełnia "po spirali" kolejnymi wyrazami ciągu arytmetycznego o danym wyrazie początkowym i zadanej różnicy ciągu. Do procedury należy przekazać: wypełnianą tablicę, wyraz początkowy i różnicę ciągu.

2	4	6	8	10
32	34	36	38	12
30	48	50	40	14
28	46	44	42	16
26	24	22	20	18

8. Dane są deklaracje:

```
const
    max = 10;
type
    tablica = array[1..max, 1..max] of integer;
var
    T: tablica;
```

Zakładamy, że zmienna typu tablica jest wypełniona przypadkowymi wartościami. Proszę napisać procedurę, która dla zmiennej typu tablica zwraca współrzędne i, j elementu T[i, j] dla którego suma elementów w "ukośnych rzędach" jest największa.

9. Dane są definicje:

```
const
    max1 = 100; max2 = 200;
type
    wiersz = array[1..max2] of integer;
    table = array[1..max1] of wiersz;
```

Typ table opisuje tablicę zawierającą max1 wierszy po max2 elementów. Proszę napisać funkcję, która dla zmiennej typu table zwraca ilość wierszy, w których wartości elementów stanowią ciągi arytmetyczne.

10. Dane są definicje:

```
const
    max1 = 100;
    max2 = 2000;
type
    table = array[1..max1, 1..max2] of real;
```

Proszę napisać procedurę, która dla zmiennej typu table zwraca numer wiersza i kolumny elementu, dla którego suma otaczających go elementów jest najmniejsza.

11. Dane są następujące definicje:

```
const
    max = 10;
var
    tab: array[1..max] of real;
```

Proszę napisać program, który wczytuje wprowadzony z klawiatury ciąg dodatnich liczb rzeczywistych zakończony wartością -1, będącą znacznikiem końca ciągu i umieszcza max największych elementów w tablicy tab. Można założyć, że ciąg zawiera co najmniej max liczb.

12. Dany jest typ strukturalny opisujący liczbę zmiennoprzecinkową:

```
const
    max = 200;
type
    bit = 0..1;
    bigfloat = record
        m: array[0..max] of bit;
        c: integer;
    end;
```

Mantysa jest reprezentowana w kodzie U2, kropka położona jest na prawo od mantysy. Mantysa jest znormalizowana, gdy najbardziej znaczący bit mantysy jest równy 1. Cecha dana jest w postaci liczby typu integer. Zmienna typu integer zajmuje 2 bajty. Proszę napisać procedurę dodającą dwie znormalizowane liczby typu bigfloat. Procedura powinna sygnalizować wystąpienie nadmiaru zmiennoprzecinkowego.

Zadania z wskaźnikami

1. Dane są deklaracje opisujące listę jednokierunkową:

```
type
  pwezel = ^wezel;
  wezel = record
    val: integer;
    next: pwezel;
  end;
```

Proszę napisać procedurę, która w liście wskazywanej przez zmienną typu pwezel usuwa powtarzające się elementy. Można założyć, że lista jest niepusta.

2. Dane są definicje opisujące listę dwukierunkową:

```
type
  pwezel = ^wezel;
  wezel = record
    next: pwezel;
    prev: pwezel;
  end;
```

Dana jest zmienna globalna wskazująca na początek listy. Proszę napisać procedurę, która wstawia do powyższej listy nowy element typu wezel na n-tą pozycję. Można założyć, że lista posiada co najmniej n elementów. Do procedury należy przekazać: wskaźnik na początek listy, wstawiany element oraz pozycję, na której należy wstawić nowy element.

3. Dane są definicje opisujące listę dwukierunkową:

```
type
  pwezel = ^wezel;
  wezel = record
    next: pwezel;
    prev: pwezel;
  end;
```

Dana jest zmienna globalna wskazująca na początek listy. Proszę napisać procedurę, która z powyższej listy usuwa z n-ty element, jeżeli lista posiada co najmniej n elementów; jeżeli lista posiada mniej niż n elementów procedura nie robi nic. Do procedury należy przekazać wskaźnik na początek listy oraz numer elementu do usunięcia.

4. Dane są definicje opisujące listę jednokierunkową:

```
type
  pwezel = ^wezel;
  wezel = record
    nr: integer;
    next: pwezel;
  end;
```

Pojedyncza, niepusta lista zakończona jest cyklem. Proszę napisać funkcję, która zwraca dla danej listy ilość elementów w cyklu. Do funkcji należy przekazać wskaźnik na początek listy. Można założyć, że ilość elementów w liście nie przekracza 100 **[Można rozważyć także przypadek, kiedy nie znamy a priori ograniczenia ilości elementów w liście]**.

5. Dane są definicje:

```
type
  pwezel = ^wezel;
  wezel = record
    val: integer;
    next: pwezel;
  end;
```

Globalna zmienna typu pwezel wskazuje na listę cykliczną. Proszę napisać procedurę, usuwającą nadmiarowe (tj. o powtarzających się wartościach) elementy z listy. Można założyć, że lista zawiera co najmniej 2 różne elementy.

6. Dane są deklaracje opisujące listę jednokierunkową:

```
type
```

```

pwezel = ^wezel;
wezel = record
    val: integer;
    next: pwezel
end;

```

Pojedyncza lista zawiera niepowtarzające się liczby naturalne uporządkowane rosnąco. Proszę napisać procedurę, która dwie niepuste listy wejściowe przekształca w jedną listę wynikową tak, że elementy listy wynikowej stanowią sumę mnogościową list wejściowych. Listy wejściowe są usuwane z pamięci.

7. Dane są definicje opisujące listę jednokierunkową:

```

type
    pwezel = ^wezel;
    wezel = record
        val: integer;
        next: pwezel
    end;

```

Pojedyncza lista zawiera niepowtarzające się liczby naturalne uporządkowane rosnąco. Proszę napisać funkcję, która dwie niepuste listy wejściowe przekształca w jedną listę wynikową tak, że elementy listy wynikowej stanowią sumę mnogościową list wejściowych. Listy wejściowe są usuwane z pamięci. Do funkcji należy przekazać wskaźniki na początki list wejściowych, funkcja powinna zwrócić wskaźnik do listy wynikowej.

8. Dane są definicje:

```

const
    max = 500;
type
    tablica = array[1..max, 1..max] of integer;

```

W zmiennych typu tablica większość (np. 95%) elementów jest równa 0. Proszę zaproponować strukturę dynamiczną (wskaźnikową) przechowującą w sposób efektywny (pod względem wykorzystania pamięci) zawartość zmiennej typu tablica.

- o Proszę napisać procedurę przekształcającą zmienną typu tablica na zaproponowaną strukturę;
- o Proszę napisać funkcję, która korzystając z zaproponowanej struktury zwraca wartość wybranego elementu (określonego przez parę indeksów (i, j)).

9. Dane są deklaracje opisujące listę jednokierunkową:

```

type
    pwezel = ^wezel;
    wezel = record
        val: integer;
        next: pwezel
    end;

```

Pojedyncza lista zawiera niepowtarzające się liczby naturalne uporządkowane rosnąco. Proszę napisać procedurę, która dwie niepuste listy wejściowe przekształca na jedną listę wynikową tak że elementy listy wynikowej stanowią iloczyn mnogościowy list wejściowych. Listy wejściowe są usuwane z pamięci.

10. Dane są deklaracje opisujące listę jednokierunkową:

```

type
    pwezel = ^wezel;
    wezel = record
        potega: integer;
        wspolcz: integer;
        next: pwezel
    end;

```

Lista reprezentuje wielomian o współczynnikach całkowitych. Elementy w liście ułożone są według rosnących potęg. Proszę napisać procedurę obliczającą różnicę dwóch dowolnych

wielomianów. Wielomiany reprezentowane są przez wyżej opisane listy. Procedura powinna zwracać wskaźnik do nowo utworzonej listy reprezentującej wielomian wynikowy. Listy wejściowe powinny pozostać niezmienione.

11. Dane są deklaracje opisujące listę jednokierunkową:

```
type
    pwezel = ^wezel;
    wezel = record
        next: pwezel
    end;
```

Proszę napisać procedurę, która w liście wskazywanej przez zmienną typu pwezel odwraca kolejność jej elementów. Można założyć, że lista jest niepusta.

Zadania różne

1. Proszę napisać program rozwiązujący problem 8 hetmanów ustawionych na szachownicy tak, aby żaden nie atakował innego.
2. Proszę napisać program, który na szachownicy "odwiedzi" wszystkie pola tylko raz, poruszając się ruchem konika szachowego.
3. Proszę napisać procedurę, otrzymującą jako argument tablicę liczb typu Integer wraz z jej długością oraz liczbę naturalną N mniejszą lub równą długości tablicy. Efektem działania procedury powinno być wypisanie na standardowe wyjście wszystkich N-elementowych kombinacji liczb z tablicy w oddzielnych wierszach. Można przyjąć, że liczby w tablicy się nie powtarzają, a jej długość jest mniejsza lub równa stałej Max.

[Przykładowe rozwiązanie nr 1](#)

[Przykładowe rozwiązanie nr 2](#)

4. Proszę napisać procedurę, otrzymującą jako argument tablicę liczb typu Integer wraz z jej długością oraz liczbę naturalną N mniejszą lub równą długości tablicy. Efektem działania procedury powinno być wypisanie na standardowe wyjście wszystkich N-elementowych wariacji bez powtórzeń liczb z tablicy w oddzielnych wierszach. Można przyjąć, że liczby w tablicy się nie powtarzają, a jej długość jest mniejsza lub równa stałej Max.

[Przykładowe rozwiązanie nr 1](#)

[Przykładowe rozwiązanie nr 2](#)

5. Proszę napisać definicje i deklaracje odpowiednich typów i zmiennych tak, aby poniższe odwołania (jeżeli to możliwe) były poprawne w Pascalu.

```
a[3][4].b
a.b[6]
a^b.c
c(b.e)
a[b[3]]
b.c(4)
a[5]^b[6]
```