

Podstawy Informatyki

Ćwiczenia tablicowe Nr 3

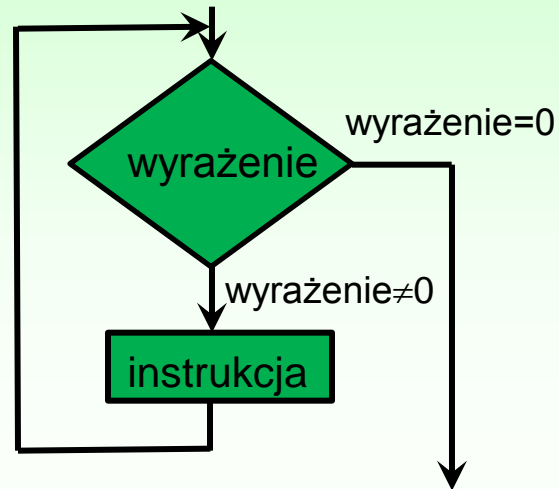
Plan dzisiejszych zajęć:

- 1. Algorytmy i programy z zastosowaniem pętli while i do while**
- 2. Tematyka następnych zajęć**

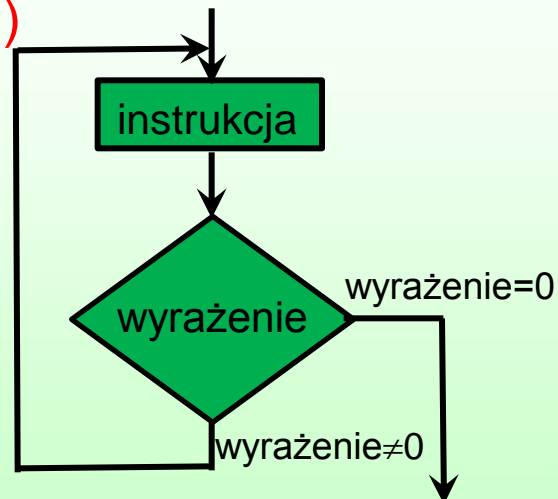


Instrukcje pętli:

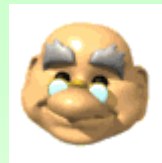
while (wyrażenie) instrukcja



do instrukcja **while** (wyrażenie)



Tematy zadań do opracowania z pętli:



1. Napisz program w języku ANSI C, który wczytuje dowolne liczby całkowite i wypisuje na monitorze najmniejszą z tych liczb. Program kończy działanie, gdy wprowadzimy 0 (0 jest traktowane wyłącznie jako warunek końca, a nie jako jedna z wprowadzanych liczb).
2. Napisz program w języku ANSI C, który wczytuje dowolne znaki z klawiatury aż do momentu naciśnięcia znaku '*'. Wtedy program powinien zapytać użytkownika czy chce przerwać wczytywanie. Po naciśnięciu znaku 't' lub 'T' wczytywanie ma być przerwane, natomiast po naciśnięciu innego klawisza wczytywanie powinno być kontynuowane.
3. Napisz program w języku ANSI C, który wczytuje dowolne liczby całkowite i wypisuje na monitorze sumę wprowadzonych liczb nieujemnych. Program kończy działanie, gdy wprowadzimy liczbę podzielną przez 13.
4. Napisz program w języku ANSI C, który wczytuje dowolne liczby całkowite i wypisuje na monitorze sumę wprowadzonych liczb nieparzystych. Program kończy działanie, gdy obliczona suma będzie podzielna przez 100.
5. Napisz program w języku ANSI C, który wczytuje liczby zmiennoprzecinkowe z klawiatury, sprawdza, czy są większe od zera (pozostałe pomija) i wypisuje je na monitorze do momentu, dopóki ich iloczyn nie przekroczy 500.

Tematy zadań do opracowania z pętli:

6. Napisz program w języku ANSI C, który wczytuje z klawiatury liczby całkowite (i wypisuje na monitorze) aż do momentu natrafienia na liczbę podzielną przez 11 lub 13.

7. Napisz program w języku ANSI C, który wczytuje z klawiatury dowolne znaki aż do momentu, gdy kolejno wprowadzone znaki będą gwiazdką i kropką.

8. Napisz program w języku ANSI C, który wczytuje i wypisuje z klawiatury dowolne znaki (pojedynczo) aż do momentu, gdy dwa kolejno wprowadzane znaki powtórzą się.

9. Napisz program w języku ANSI C, który wczytuje po dwie dowolne liczby całkowite, porównuje je i wypisuje większą. Wczytywanie kończy się, gdy wprowadzimy w kolejnej dwójce dwie jednakowe liczby.

10. Napisz program w języku ANSI C, który na zmianę wczytuje i wypisuje liczby zmiennoprzecinkowe i znaki, dopóki iloczyn wczytanych liczb zmiennoprzecinkowych nie przekroczy 350.

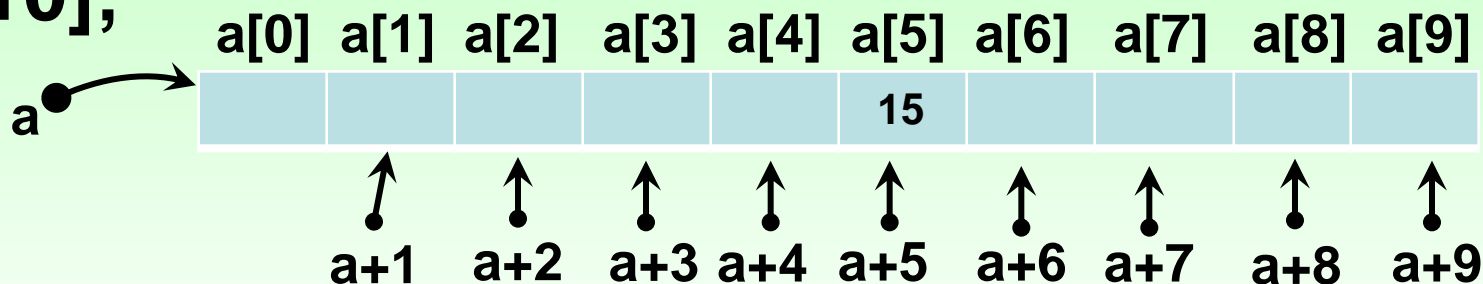
Następne ćwiczenia:

- **Kartkówka z instrukcji pętli while i do while**
- **Rozwiązywanie zadań z zastosowaniem tablic (wykorzystanie pętli for).**



I. Typ złożony: tablice

```
int a[10];
```



```
a[5]=15;
```

zapis `a[0]` jest równoważny `*a`
zapis `a[1]` jest równoważny `*(a+1)`
zapis `a[2]` jest równoważny `*(a+2)`

.....

zapis `a[9]` jest równoważny `*(a+9)`

zapis `&a[0]` jest równoważny `a`
zapis `&a[1]` jest równoważny `a+1`
zapis `&a[2]` jest równoważny `a+2`

.....

zapis `&a[9]` jest równoważny `a+9`

czyli jeżeli `x` jest zmienną typu `int`:
`x=a[i]` jest równoważne `x=*(a+i)`

Podsumowując:

`a[k]` i `*(a+k)` to wartości w `k`-tym elemencie tablicy;
`&a[k]` i `a+k` to adresy `k`-tego elementu tablicy



I. Typ złożony: tablice

Przykłady deklaracji tablic:

float liczby[100]; - deklaracja 100-elementowej tablicy liczb zmiennoprzecinkowych

char znaki[25]; - deklaracja 25-elementowej tablicy znaków

int macierz[25][10]; - deklaracja 250- elementowej tablicy dwuwymiarowej liczb int

int *tablica[10]; - deklaracja 10-elementowej tablicy wskaźników do liczb typu całkowitego

char *tekst; - deklaracja wskaźnika do znaku

Przykłady nadawania wartości elementom tablic:

liczby={1,2,10,-5,0,8};

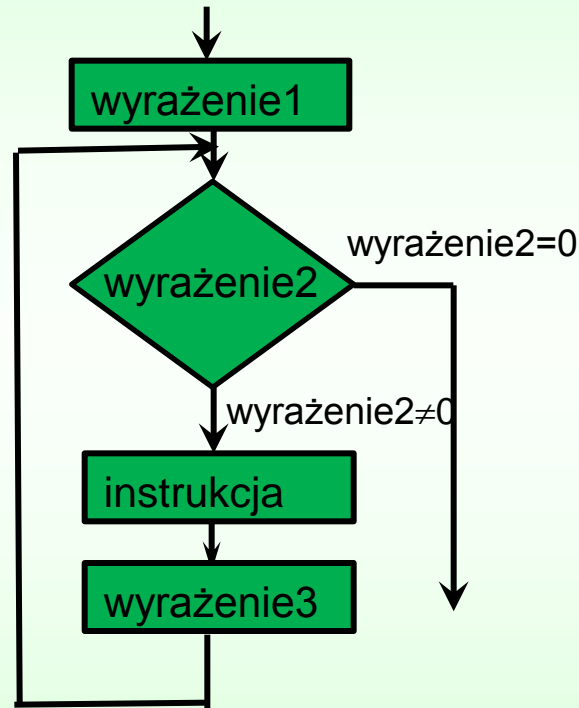
tekst=„wyraz1 wyraz2”;

macierz={{1,2,3},{4,5,6},{7,8,9}};



Instrukcja pętli for:

for (wyrażenie1; wyrażenie2; wyrażenie3) instrukcja



wyrażenie1 – określa warunki początkowe pętli

wyrażenie2 – określa warunek wykonania pętli

wyrażenie3 – wykonywane (za każdym razem) po instrukcji w pętli.

Tematy zadań do opracowania z tablic:



1. Napisz program, który przesuwą w lewo o jedną pozycję zawartość tablicy 10-cio elementowej liczb całkowitych tzn. dla elementów $i=0, \dots, 8$ $t[i]$ po wykonaniu programu ma mieć wartość $t[i+1]$, a element $t[9]$ wartość $t[0]$. Program ma używać tylko jednej tablicy.

2. Napisz program, który z dwóch 10-cio elementowych posortowanych tablic liczb zmiennoprzecinkowych tworzy trzecią również posortowaną. Należy wykorzystać fakt posortowania tablic 10-cio elementowych (do trzeciej tablicy, kolejne elementy mają być wstawiane od razu we właściwym miejscu tej tablicy).

3. Napisz program, który z dwóch 10-cio elementowych tablic znakowych tworzy trzecią złożoną na zmianę z elementów to jednej, to drugiej tablicy.

4. Napisz program, który oblicza iloczyn tylko tych elementów całkowitoliczbowej tablicy 10-cio elementowej, które są parzyste.

5. Napisz program, który posortowaną tablicę liczb całkowitych sortuje w kierunku przeciwnym. Należy wykorzystać początkowe posortowanie tablicy. W programie można użyć tylko jedną tablicę.

6. Napisz program, który zamienia liczbę całkowitą dziesiętną na system dwójkowy (bez użycia funkcji matematycznych dostępnych w C i formatu binarnego) .

7. Napisz program, który wpisuje do kolejnych elementów tablicy 20-to elementowej kolejne potęgi liczby n , począwszy od n^0 . Liczbę n należy wczytać z klawiatury. Potęgowanie należy zrealizować za pomocą wielokrotnego mnożenia.

Tematy zadań do opracowania z tablic:

8. Napisz program, który wstawia dowolną liczbę zmiennoprzecinkową do posortowanej tablicy 10-cio elementowej o elementach -5,5; -1,2; 0; 0; 1,4; 3,5; 6,8; 10; 10; 14,5. tak, aby tablica była nadal posortowana bez sortowania całego wektora. Należy zadeklarować tablicę 11-to elementową, żeby wstawiany element się zmieścił.

9. Napisz program, który oblicza średnią arytmetyczną elementów podzielnych przez 15 w 10-cio elementowej tablicy liczb całkowitych.

10. Napisz program, który z dwóch 10-cio elementowych tablic liczbowych tworzy trzecią w ten sposób, że jej kolejne elementy zawierają na zmianę dwa elementy z jednej tablicy i dwa z drugiej. Jeżeli elementy w jednej z tablic skończą się, pozostałe elementy z drugiej są przepisywane.