

Przechowywanie danych w tablicach

Tablica jest to ciąg komórek pamięci, które mają jedną nazwę i jednakowy typ. Tablice są nazywane tablicami statycznymi, ponieważ liczba przechowywanych przez nie elementów, jak również zużywana pamięć są stałe i ustalone w trakcie kompilacji.

Deklaracja tablicy:

typ_elementu nazwa_tablicy [wymiar_1] [wymiar_2]...[wymiarN]

```
//tablica
//deklaracja
int tab[10];

//deklaracja wraz z inicjalizacją
int tabInt[5] = { 1,2,3,4,5 };
int MyNumbers[5] = { 0 }; //wszystkie elementy tablicy przyjmują wartość 0
int tab1[100] = { 23,15 }; //zainicjalizowane elementy: 23, 15. 0, 0, 0 ... n, n -
rozmiar tablicy
float tabFloat[] = { 2.3, 4.5,1.34 };

//deklaracja tablicy znakowej
char tabChar[10];

// deklaracja tablicy dwuwymiarowej o 5 wierszach i 3 kolumnach
int tab2D[5][3];
```

Wielkość tablicy (tzn. liczbę znajdujących się w niej elementów) można zdefiniować w postaci stałej, która następnie będzie użyta w definicji tablicy:

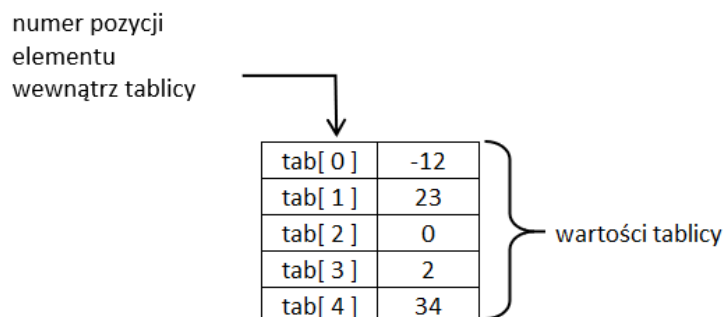
```
const int rozmiar = 5;
int tabInt[rozmiar];
```

Jeżeli znane są początkowe wartości elementów tablicy, wtedy można pominąć liczbę wskazującą wielkość tablicy, np.:

```
int tabInt[] = { 2011, 2052, -525 }; //utworzono tablicę zawierającą 3 liczby całkowite:
2011, 2052 i -525.
```

Indeksowanie tablic rozpoczynamy od 0.

Tablica jednowymiarowa



Odwoływanie się do elementów tablicy:

```
int tabInt[10];

tabInt[0] = 12;
tabInt[1] = 3;
tabInt[2] = tabInt[1] - tabInt[0];
tabInt[3] = tabInt[4] * 2; // błąd nie istnieje element tabInt[4];

printf("%d ", tab[0]);
printf("\n%d ", tab[1]);
printf("\n%d ", tab[2])
```

Wyświetlanie elementów tablicy:

```
//wyświetlanie elementów tablicy
for (int i = 0; i < 10; i++)
{
    tab[i] = i;
    printf("%d ", tab[i]);
}
```

Zadanie 1. Napisz program wprowadzający do 10-elementowej tablicy wartości: 3, 6, 9, ...

Zadanie 2. Do programu z zadania 1 dodaj instrukcje wyświetlające co drugą wartość z tablicy.

Zadanie 3. Wypełnij n-elementową tablicę, zaczynając od ostatniej komórki tablicy, wartościami a, a-5, a-10, Wartość całkowitą a podaje użytkownik.

Zadanie 4. Napisz program znajdujący minimalną wartość w tablicy n-elementowej, elementy tablicy są losowane z przedziału [-23, 50].

Zadanie 5. Użytkownik podaje 2 pierwsze wartości tablicy. Wypełnij pozostałe komórki tablicy wartościami, które stanowią sumę wartości z dwóch poprzedzających komórek tablicy. np.: 3, 6, ?, ?, ?
-> 3, 6, 9, 15, 24.

Zadanie 6. Dana jest tablica 20 elementowa. Pierwsza połowa tablicy jest wypełniona. Napisz program, który wypełni drugą połowę tablicy poprzez kopiowanie kolejnych wartości z pierwszej połowy tablicy.

Tablica wielowymiarowa

```
//deklaracja i inicjalizacja tablicy dwuwymiarowej
int tab2D[2][3] = { 1,2,3,4,5,6 };
```

i-ty wiersz
j-ta kolumna

	j = 0	j = 1	j = 2
i = 0	6	-23	16
i = 1	332	0	2
i = 2	2	3	56
i = 3	-123	21	23
i = 4	123	2	36

tab[0][0]

tab[2][2]

tab[4][0]

Odwołanie do wartości z poszczególnych komórek odbywa się poprzez podanie nazwy tablicy, numeru wiersza (indeks wiersza), numeru kolumny (indeks kolumny).

```
int tab2D[2][3];

tab2D[0][0] = 23;
tab2D[0][1] = -12;
tab2D[0][2] = tab2D[0][0] + tab2D[0][1];
int x = tab2D[0][0] * 2
```

Zadanie 7. Napisz program obliczania sumy elementów przekątnej głównej macierzy.

Zadanie 8. Napisz program dodawania dwóch macierzy.