

# Laboratorium nr 5

Temat: Tablice statyczne.

Zakres laboratorium:

- tablice statyczne
- inicjalizowanie tablic
- tablice znakowe
- zadania laboratoryjne

# Tablice statyczne

**Tablica** – ciąg obiektów tego samego typu, które zajmują ciągły obszar w pamięci, np.

```
int tablica[20];           //tablica liczb typu int
```

```
int tablica[3][4];       //tablica liczb typu int
```

```
char zdanie[80];        //tablica znakowa char
```

```
float numery[20];       //tablica liczb typu float
```

```
int *tabl_wskaz[12];    //tabl wskaźników do int
```

**Numeracja elementów tablicy w C++ zaczyna się od zera!!!**

**NAZWA TABLICY jest równocześnie ADRESEM JEJ  
ZEROWEGO ELEMENTU!!!**

# Inicjalizowanie tablic

**Inicjalizowanie tablic jednowymiarowych** – wartości początkowe elementom tablicy można nadać w różny sposób:

**- za pomocą przypisania z wykorzystaniem pętli**

```
int tablica[10]; //tablica 10 liczb typu int
for(int i=0;i<10;i++)
    tablica[i]=0; //inicjalizacja w pętli

for(int i=0;i<10;i++)
    cin>>tablica[i]; //wczytanie elementów z klawiatury
```

**- inicjalizacja w momencie definicji tablicy**

```
int tab1[5]={0}; //wszystkie elementy zerowe
int tab2[5]={2,5,6}; //elementy 3 i 4 zerowe
int tab3[5]={1,2,3,4,5}; //jawna inicjalizacja elementów
int tab4[ ]={1,2,3,4}; //liczba elementów tablicy równa 4
int tab5[5]={0,1,2,3,4,5}; //błąd składni
const int rozmiar=5; //rozmiar tablicy - stała symboliczna
int tab6[rozmiar]={0}; //rozmiar wcześniej zdefiniowany
```

**- przypisanie każdego elementu tablicy z osobna**

```
int tab[3]; //tablica 3 liczb typu int
tab[0]=1; tab[1]=4; tab[2]=-3 //jawna inicjalizacja (w jednej linii)
```

**Inicjalizowanie tablic dwuwymiarowych** – wartości początkowe elementom tablicy można nadać w różny sposób:

**- za pomocą przypisania z wykorzystaniem pętli**

```
int tablica[10][10];           //tablica 20 liczb typu int
for(int i=0;i<10;i++)
    for(int j=0;j<10;j++)
        tablica[i][j]=0;      //inicjalizacja w pętli

for(int i=0;i<10;i++)
    for(int j=0;j<10;j++)
        cin>>tablica[i][j];  //wczytanie elementów z klawiatury
```

**- inicjalizacja w momencie definicji tablicy**

```
int tab1[2][2]={0};           //wszystkie elementy zerowe
int tab2[2][2]={ {1}, {3,4} }; //element tab2[0][1] zerowy
int tab3[2][2]={ {1,2}, {3,4} }; //wszystkie elementy jawnie zainicjalizowane
int tab4[2][2]={ {1,2,3,4} };  //wszystkie elementy jawnie zainicjalizowane
```

**- przypisanie każdego elementu tablicy z osobna**

```
int tab[2][2];                //tablica 4 liczb typu int
tab[0][0]=1; tab[0][1]=2;     //jawna inicjalizacja
tab[1][0]=3; tab[1][1]=4;     //jawna inicjalizacja
```

# Tablice znakowe

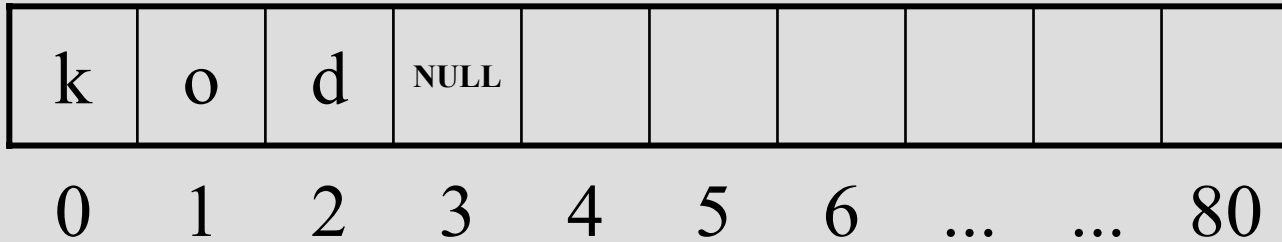
**Tablica znakowa** – tablica do przechowywania znaków (np. liter).

**String** – ciąg znaków ASCII zakończony znakiem NULL (znak o kodzie 0).

```
char zdanie[80]; //tablica znakowa char

char zdanie[80]={"kod"}; //znak NULL automatycznie dopisany,
char zdanie[ ]={"kod"}; //string poprawnie zakończony,
//jest znak NULL

char zdanie[80]={'k','o','d'}; //reszta inicjalizowana zerami,
//a więc jest znak NULL -
//string jest poprawnie zakończony
```



```
char zdanie[ ]={'k','o','d'}; //nie ma znaku NULL, nie jest to
//błąd, gdy litery nie mają być
//używane jako ciąg znaków - czyli
//string - ale jako luźne litery do
//innych celów
```



# Zadania laboratoryjne