

## 5 Tablice

- *Tablica* jest zestawem obiektów (zmiennych) *tego samego typu*, do których można się odwołać za pomocą wspólnej nazwy.
- Obiekty składowe tablicy noszą nazwę *elementów tablicy*. Dostęp do nich jest możliwy poprzez podanie położenia elementu w tablicy. Ten rodzaj dostępu nazywa się indeksowaniem (ang. *indexing, subscripting*).

### 5.1 Tablice jednowymiarowe

- Składnia deklaracji tablicy jednowymiarowej:

```
<typ> <nazwa_tablicy>[<wymiar>;
```

gdzie:

*<typ>* tablicy określa typ wszystkich jej elementów.

*<wymiar>* (ang. *dimension*) tablicy określa liczbę elementów, które można zapisać w tablicy.

- Przykład: deklaracja tablicy dziesięciu obiektów typu `int` ma postać  
`int t[10];`
- Wymiar musi być wyrażeniem **stałym typu całkowitego**, tzn. takim, które może obliczyć kompilator.
- Przykłady:

```
int wyniki[25] /*tablica przechowująca 25 liczb typu int */  
char alfabet[26]; /* tablica przechowująca 26 znaków */  
  
const int rozmiar_bufora=80;  
char bufor_we[rozmiar_bufora];
```

#### Dostęp do elementów tablicy

- Nazwa tablicy jest wspólną nazwą wszystkich elementów tablicy. Dostęp do elementów tablicy jest realizowany za pomocą indeksu, który wskazuje położenie danego elementu w tablicy. Indeks może być dowolnym wyrażeniem o wartości całkowitej, zawierającym zmienne i stałe całkowite.
- **Pierwszy element tablicy ma indeks równy 0. Ostatni element tablicy ma indeks równy rozmiarowi tablicy minus 1.**
- Przykład:

```
int wyniki_testu[25]; // tablica o 25 elementach typu int  
wyniki_testu[0]=3; // pierwszy element tablicy  
wyniki_testu[24]=5; // ostatni element tablicy
```

- Indeks nie musi być stałą, może być obliczany. **Niebezpieczeństwo: w języku C i C++ nie sprawdzane jest automatycznie przekroczenie zakresu indeksów tablicy. Programista musi sam zadbać o to, aby użyć właściwej wartości indeksu: nieujemnego i mniejszego od rozmiaru tablicy.**

## Rozmieszczenie tablicy w pamięci

- Każda tablica zajmuje spójny obszar pamięci. Przykład: tablica `int a[4]={5,4,8,7}`, (zakładamy, że typ `int` zajmuje 2 bajty):

element	a[0]		a[1]		a[2]		a[3]	
wartość	5		4		8		7	
adres	1000		1002		1004		1006	
komórki	1000	1001	1002	1003	1004	1005	1006	1007

- Całkowity rozmiar tablicy w bajtach można obliczyć za pomocą jednego ze wzorów:

```
razem_bajtow = sizeof(typ) * liczba_elementów_tablicy  
razem_bajtow = sizeof nazwa_tablicy
```

## 5.2 Nadawanie wartości elementom tablicy

### Inicjowanie tablicy przy deklaracji

```
int liczby[3]={1,2,3}; // inicjacja: podane wartości wszystkich  
                      // elementów  
int liczby[3]={1}; // niepełna inicjacja  
int liczby[]={1,2,3}; // automatyczne nadawanie rozmiaru:  
                      // kompilator sam obliczy rozmiar
```

- Przykład: W tablicy `dni` przechowywane są liczby dni w każdym miesiącu roku. Chcemy je wydrukować.

```
#include <iostream>  
using namespace std;  
int main()  
{  
    int dni[] = {31,28,31,30,31,30,31,31,30,31,30,31};  
    for (int i = 0; i < sizeof dni / sizeof (int); i++)  
        cout << "Miesiac " << (i+1) << " ma " << dni[i] << " dni." << endl;  
    return 0;  
}
```

### Przypisywanie wartości elementom tablicy

- Wartości można przypisać jedynie pojedynczym elementom tablicy. Przykład:

```
int wyniki_testu[20]; // deklaracja tablicy  
wyniki_testu[0]=3; // przypisanie wartości elementowi tablicy
```

- Nie można jednej tablicy bezpośrednio przypisać innej tablicy:

```
int x[] = {0,1,2};  
int y[3];  
x=y; // błąd
```

- Aby utworzyć kopię tablicy, trzeba kolejno kopiować każdy element tablicy. Przykład(kopia):

```
#include <iostream>  
using namespace std;  
int main() {  
    const int rozmiar=5;  
    int x[rozmiar], y[rozmiar];  
    int i;  
    // przypisywanie wartości elementom tablicy x
```

```
for (i=0; i<rozmiar; ++i)
    x[i]=i;
// kopiowanie tablicy x do y
for (i=0; i<rozmiar; ++i)
    y[i]=x[i];
// drukowanie tablicy y
for (i=0; i<rozmiar; ++i)
    cout << y[i] << ' ';
cout << endl;
return 0;
}
```

### Wczytywanie wartości elementów tablicy

- Wartości można wczytywać tylko do pojedynczych elementów tablicy. Przykład:

```
int wyniki_testu[20];
for (int i=0; i<3; i++)
    cin >> wyniki_testu[i];
```

## 5.3 Przykłady użycia tablic jednowymiarowych

- **Przykład 1.** Wczytać dzienne pomiary temperatury w tygodniu i obliczyć ich średnią.

```
#include <iostream>
using namespace std;
int main() {
    int const TYDZIEN=7;
    int i;
    double temperatura[TYDZIEN], suma=0 ;
    for (i=0;i<TYDZIEN;i++) {
        cin >> temperatura[i];
        suma += temperatura[i];
    }
    cout << "Srednia temperatura w tygodniu: " << suma/TYDZIEN << endl;
    return 0;
}
```

- **Przykład 2.** Wczytać do tablicy 10 liczb i wydrukować je w odwrotnej kolejności.

```
#include <iostream>
using namespace std;
int main() {
    const int MAX=10;
    int t[MAX], ile; // ile - faktyczna liczba elementów
    cout << "Podaj liczbe elementow: ";
    cin >> ile;
    if (ile < 0 || ile > MAX) {
        cout << "Liczba elementów musi być nieujemna i mniejsza od "<<MAX<<
            endl;
        return 1;
    }
    cout << "Podaj liczby:" << endl;
    /* wpisanie liczb do tablicy */
    for (int i=0; i<ile; i++)
        cin >> t[i];
    /* wyświetlenie liczb z tablicy */
    cout << "Liczby w odwrotnej kolejnosci:" << endl;
    for (int i=ile-1; i>=0; i--)
        cout << t[i] << ' ';
    cout << endl;
    return 0;
}
```

- **Przykład 3:** Obliczyć krotność występowania cyfr 0-9 we wprowadzonym tekście.

```
#include <iostream>
#include <iomanip>
using namespace std;
int main() {
    int znak,i;
    int cyfry[10];
    for (i=0;i<10;++i) cyfry[i]=0;
    while ((znak=cin.get()) != '\n')
        if (znak>='0' && znak<='9') ++cyfry[znak-'0'];
    cout << "Wprowadzone cyfry:" << endl;
    for (i=0;i<10;++i) cout << setw(3) << i;
    cout << endl;
    for (i=0;i<10;++i) cout << setw(3) << cyfry[i];
    cout << endl;
    return 0;
}
```

## 5.4 Tablice wielowymiarowe

- Deklaracja tablicy wielowymiarowej:

```
<typ> <nazwa_tablicy>[<wymiar_n>]...[<wymiar_2>][<wymiar_1>]
```

### Tablica dwuwymiarowa

- Najprostszą tablicą wielowymiarową jest tablica dwuwymiarowa.
- Deklaracja tablicy dwuwymiarowej:

```
<typ> <nazwa_tablicy>[<liczba_wierszy>][<liczba_kolumn>]
```

- Przykład deklaracji tablicy *A<sub>5,10</sub>*:

```
int A[5][10]; /* w deklaracji każdy nawias określa
               liczbę elementów,
               tablica ma 5 wierszy i 10 kolumn */
```

- Pierwszy i ostatni element tablicy A:

```
int A[0][0]; /* [wiersz][kolumna] */
int A[4][9]; /* [wiersz][kolumna] */
```

- Dostęp do elementu *a<sub>1,2</sub>*:

```
a[1][2]=5;
```

- Elementy tablicy umieszczane są w pamięci wierszami. Przykład tablicy `int a[2][3]` (zakładamy, że typ `int` zajmuje 2 bajty), w której umieszczono kolejne liczby całkowite poczynając od 0.

Element	Adres	Komórki		Zawartość
a[0][0]	1000	1000	1001	0
a[0][1]	1002	1002	1003	1
a[0][2]	1004	1004	1005	2
a[1][0]	1006	1006	1007	3
a[1][1]	1008	1008	1009	4
a[1][2]	1010	1010	1011	5

### Inicjacja tablic dwuwymiarowych

- Inicjacja tablic dwuwymiarowych jest podobna do inicjacji tablic jednowymiarowych.
- Przykład 1:

```
int a[2][3]={ {0,1,2},
              {3,4,5}};
```

- Przykład 2: Jeśli wpisujemy wszystkie wartości, to możemy pominąć nawiasy wyznaczające wiersze, musimy tylko pamiętać, że wartości początkowe nadawane są kolejnymi wierszami.

```
int a[2][3] = { 0,1,2,3,4,5 };
```

- Przykład 3: Można pominąć ostatni wymiar (ten który wymieniony jest jako pierwszy)

```
int a[][3] = { 0,1,2, 3,4,5 };
```

**Przykład 4:** Wypełnienie tablicy wartościami 1 i wyświetlenie tablicy wierszami.

```
#include <iostream>
#include <iomanip>
using namespace std;
int main() {
    const int MAXW=3;
    const int MAXK=4;
    int a[MAXW][MAXK];

    /* wpisanie liczb do tablicy */
    for (int i=0; i<MAXW; ++i)
        for (int j=0; j<MAXK; ++j)
            a[i][j]=1;

    /* wyświetlenie liczb w tablicy */
    for (int i=0; i<MAXW; ++i){
        for (int j=0; j<MAXK; ++j)
            cout << setw(2) << a[i][j];
        cout << endl;
    }
    return 0;
}
```

**Przykład 5:** Wyzerowanie głównej przekątnej w macierzy kwadratowej, przypisanie pozostałym elementom wartości 1.

```
#include <iostream>
#include <iomanip>

using namespace std;
int main() {
    const int MAX=4;
    int a[MAX][MAX], i, j;

    /* wpisanie liczb do tablicy */
    for (i=0; i<MAX; i++)
        for (j=0; j<MAX; j++)
            a[i][j]= (i==j) ? 0:1;

    /* wyświetlenie liczb w tablicy */
    for (i=0; i<MAX; ++i){
        for (j=0; j<MAX; ++j)
            cout << setw(3) << a[i][j];
        cout << endl;
    }
    return 0;
}
```

**Przykład 6.** Obliczenie i wyświetlenie sumy elementów w każdej kolumnie tablicy.

```
#include <iostream>
using namespace std; int main()
{
    const int MAXW=3;
    const int MAXK=4;
    int t[MAXW][MAXK];
    int suma;
    /* wpisanie liczb do tablicy */
    for (int i=0; i<MAXW; ++i)
    {
        cout << "Podaj " << MAXK << " elementow wiersza " << i << endl;
        for (int j=0; j<MAXK; ++j)
            cin >> t[i][j];
    }
    /* obliczenie sum w kolumnach */
    for (int j=0; j<MAXK; ++j){
        suma =0;
        for (int i=0; i<MAXW; ++i)
            suma += t[i][j];
        cout << "Suma w kolumnie " << j << " wynosi: " << suma << endl;
    }
    return 0;
}
```

**Przykład 7.** Przepisanie tablicy do wektora.

```
#include <iostream>
#include <iomanip>
using namespace std;
int main() {
    const int MAXW=3;
    const int MAXK=4;
    int t[MAXW][MAXK];
    int w [MAXW*MAXK];
    int i,j,iw;
    /* wpisanie liczb do tablicy */
    for (i=0; i<MAXW; ++i)
    {
        cout << "Podaj " << MAXK << " elementow wiersza " << i << endl;
        for (j=0; j<MAXK; ++j)
            cin >> t[i][j];
    }
    /* przepisanie tablicy do wektora */
    iw=0;
    for (i=0; i<MAXW; ++i)
        for (j=0; j<MAXK; ++j)
        {
            w[iw]= t[i][j];
            iw++;
        }
    /* wydrukowanie wektora */
    cout << "Elementy wektora" << endl;
    for (i=0; i<MAXW*MAXK; i++)
        cout << setw(3) << w[i];
    cout << endl;
    return 0;
}
```

### Przykład 8. Zamiana dwóch wybranych wierszy tablicy

```
#include <iostream>
#include <iomanip>
using namespace std;
int main()
{
    const int MAXW=3;
    const int MAXK=4;
    int t[MAXW][MAXK];
    int i,j,tymcz, w1, w2;

    /* wpisanie liczb do tablicy */
    for (i=0; i<MAXW; ++i)
    {
        cout << "Podaj " << MAXK << " elementow wiersza " << i << endl;
        for (j=0; j<MAXK; ++j)
            cin >> t[i][j];
    }
    cout << "Ktore wiersze zamienic?";
    cin >> w1 >> w2;
    // pominiete sprawdzenie, czy uzytkownik podal numery wierszy z zakresu

    /* Zamiana wierszy */
    for (j=0; j<MAXK; j++)
    {
        tymcz=t[w1][j];
        t[w1][j]= t[w2][j];
        t[w2][j]=tymcz;
    }

    cout << "Tablica po zamianie" << endl;
    for (i=0; i<MAXW; i++)
    {
        for (j=0; j<MAXK; j++)
            cout << setw(3) << t[i][j];
        cout << endl;
    }
    return 0;
}
```