

Programowanie strukturalne - wykład 6

Tablice jednowymiarowe - cd.

Wskaźniki do tablic

Nazwa tablicy jest równocześnie adresem jej pierwszego elementu.

```
#include <stdio.h>
#include <stdlib.h>
#define ROZMIAR 3

int main()
{
    int tab[ROZMIAR] = {4,0,-2};
    printf("%i",tab==&tab[0]);
    return 0;
}
```

Równoważne:

```
tab + 3 == &tab[3] // ten sam adres  
*(tab + 3) == tab[3] // ta sama wartosc
```

Uwaga na nawiasy:

```
*(tab +2) // wartosc elementu tablicy o indeksie 2  
*tab +2 // 2 dodane do wartosci elementu o indeksie 0
```

Przekazywanie tablic do funkcji

Równoważnie

```
int suma(int *tab, int n)
{
    // tutaj kod
}
```

```
int suma(int tab[], int n)
{
    // tutaj kod
}
```

Uwaga: nie możemy wprost odczytać rozmiaru(!).

Działania na wskaźnikach

- ▶ Przepisanie - przypisywana wartość może być tablicą albo zmienną poprzedzoną operatorem adresu (&), ewentualnie innym wskaźnikiem.

```
#include <stdio.h>
int main(void)
{
    int tab[4]={1,-3,4,2};
    int *wsk1, *wsk2;
    wsk1=tab;
    wsk2=&tab[2];
}
```

- ▶ Pobranie wartości (dereferencja, wyłuskiwanie) - *
- ▶ Pobranie adresu wskaźnika - &

- ▶ Dodawanie liczb całkowitych do wskaźnika — za pomocą operatora + można dodać liczbę całkowitą do wskaźnika albo wskaźnik do liczby całkowitej. Wtedy liczba całkowita jest mnożona przez liczbę bajtów zajmowaną przez wartość wskazywanego typu, a wynik jest dodawany do pierwotnej wartości adresu.

```
#include <stdio.h>
int main(void)
{
    int tab[4]={1,-3,4,7};
    int *wsk1, *wsk2;
    wsk1=tab;
    wsk2=wsk1+3;
    printf("%p %p\n",wsk1,wsk2);
    printf("%d %d\n",*wsk1,*wsk2);
    return 0;
}
```


- ▶ Zwiększenie wskaźnika (inkrementacja). Można je uzyskać przez zwykłe dodawanie lub za pomocą operatora inkrementacji. Zwiększenie wskaźnika do elementu tablicy sprawia, iż wskazuje on kolejny element.

```
#include <stdio.h>
int main(void)
{
    int tab[4]={1,-3,4,7};
    int *wsk1, *wsk2;
    wsk1=tab;
    printf("%p %d\n",wsk1,*wsk1);
    wsk1++;
    printf("%p %d\n",wsk1,*wsk1);
    return 0;
}
```

- ▶ Odejmowanie liczby całkowitej od wskaźnika — w tym celu można użyć operatora `-`; wskaźnik musi być pierwszym z operandów. Liczba całkowita jest mnożona przez liczbę bajtów zajmowaną przez wartość wskazywanego typu, a wynik jest odejmowany od pierwotnego adresu.
- ▶ Zmniejszenie (dekrementacja) wskaźników - analogicznie.

- ▶ Odejmowanie. Możliwe jest znalezienie różnicy między dwoma wskaźnikami. Działanie to jest wykonywane zwykle na wskaźnikach do elementów tej samej tablicy — w celu określenia, jak daleko od siebie się znajdują. Wynik jest wyrażony w jednostce o rozmiarze typu.

```
#include <stdio.h>
int main(void)
{
    int tab[4]={1,-3,4,7};
    int *wsk1, *wsk2;
    wsk1=tab;
    wsk2=&tab[3];
    printf("%d\n",wsk2-wsk1);
    return 0;
}
```

- ▶ Porównanie — możemy używać operatorów relacyjnych, aby porównać wartości dwóch wskaźników tego samego typu.

```
#include <stdio.h>
int main(void)
{
    int tab[4]={1,-3,4,7};
    int *wsk1, *wsk2;
    wsk1=tab;
    wsk2=&tab[3];
    printf("%d\n",wsk2>wsk1);
    return 0;
}
```

Bibliografia

- ▶ Richard Reese, Wskaźniki w języku C, Wydawnictwo Helion 2014.
- ▶ <https://pl.wikibooks.org/wiki/C/Wska%C5%BAniki>, dostęp online 15.03.2020.
- ▶ http://wazniak.mimuw.edu.pl/index.php?title=Wst%C4%99p_do_programowania_w_j%C4%99zyku_C/Wska%C5%BAniki, dostęp online 15.03.2020.
- ▶ https://pl.wikibooks.org/wiki/C/Wska%C5%BAniki_-_wi%C4%99cej, dostęp online 15.03.2020.
- ▶ Stephen Prata, Język C. Szkoła programowania. Wydanie VI, Wyd. Helion, 2016.
- ▶ <https://pl.wikibooks.org/wiki/C/Tablice>, dostęp online 20.03.2020.
- ▶ https://pl.wikibooks.org/wiki/C/Tablice_-_wi%C4%99cej, dostęp online 20.03.2020.