

Základy programování 1

Mgr. Markéta Trnečková, Ph.D.



Palacký University, Olomouc

1. prvek	2. prvek	3. prvek	4. prvek
----------	----------	----------	----------

- **index**: 0, 1, ...
- `typ jmeno[velikost];`
- `typ jmeno[velikost]={p1, p2, ..., pn};`

Example

```
/* Neinicializovane pole velikosti 6 */  
int pole1[6];  
  
/* Pole o velikosti 10, se 3 inicializovanymi prvky*/  
int pole2[10] = {1, 2, 3};  
  
/* Pole o velikosti 4 se vsemi inicializovanymi prvky*/  
int pole3[] = {1, 2, 3, 4};
```

- `jmeno[index];`

Example (Co je špatně?)

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

```
pole[2] = 10;
```

```
pole[3] = pole[1] + pole[2];
```

```
pole[1] = pole[5];
```

```
pole[7] = 10;
```

Example (Co je šatně?)

```
int pole[20];  
int i;  
  
for(i = 0; i < 20; i++)  
{  
    pole[i] = 2*i;  
}
```

Example

```
#include <stdio.h>

int main()
{
    int pole[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
    int hledany = 4;
    int index;
    int i;

    index = -1;
    for(i = 0; i < 10; i++)
    {
        if(pole[i]==hledany)
        {
            index = i;
            break;
        }
    }
}
```

- Upravte předchozí příklad, aby se hodnoty pole a hledaný výraz načítaly od uživatele.
- Vypište index hledaného prvku v poli.

- 1 Upravte kód pro nalezení prvku v poli, pokud můžeme předpokládat, že prvky v poli jsou setříděny od nejmenšího.
- 2 Napište program, který vypočte průměr pole desetinných čísel.
- 3 Napište program, který zjistí, zda jsou dvě pole stejná (tj. mají na všech indexech stejné prvky). Můžeme předpokládat, že pole mají stejný počet prvků.
- 4 Upravte předchozí program tak, aby zjistil, zda pole obsahují stejné hodnoty (mohou být na různých indexech). Pozor, pole mohou obsahovat nějakou hodnotu víckrát!
- 5 Napište program, který sečte všechna lichá čísla z pole.
- 6 Napište program, který otočí pořadí prvků v poli.
- 7 Napište program, který nahradí všechny výskyty hodnoty a v poli hodnotou b .

Pole inicializujte už v kódu!