

Základy programování (IZP)

4. počítačové cvičenie

Brno University of Technology, Faculty of Information Technology
Božetěchova 1/2. 602 00 Brno - Královo Pole
ilazur@fit.vut.cz



Formátovanie zdrojového kódu

```
if (i != 0) { i += 1 }  
if (i != 0) {  
    i += 1;  
}
```

Pomenovanie identifikátorov

```
char variant_dlheho_id;  
char variantDlhehoID;
```

Konštanty

```
const KONSTANTA;
```

Zvoľte si jednu konvenciu a dodržiavajte ju

Deklarácia, definícia, inicializácia premennej, priradenie do premennej, L-hodnota, R-hodnota

```
int i;  
char s[] = "Hello";  
char c1 = '\0', c2;  
char s2[10], c;  
  
i = 2;  
c2 = s[i];
```

Deklarácia, definícia, inicializácia premennej, priradenie do premennej, L-hodnota, R-hodnota

```
int i; // deklaracia a definicia
char s[] = "Hello"; // deklaracia a definicia a
                    inicializacia premennej
char c1 = '\\0', c2;
char s2[10], c;

i = 2; // priradenie do premennej
c2 = s[i]; // L-hodnota = R-hodnota
```

Deklarácia a definícia funkcie

```
int my_abs (int x) ;
```

```
int my_abs (int x)
{
    if (x < 0)
        return -x;
    else
        return x;
}
```

Obsahujú zadané stringy iba písmená?

- 1 Načítajte zo STDIN 2 stringy, ktorých dĺžka je maximálne 100 znakov
- 2 Pre každý string vypíšte, či obsahuje iba písmená

Obsahujú zadané stringy iba písmená?

```
#include <stdio.h>

int main() {
    char name(101);
    char surname(101);

    printf("Please specify your name: ");
    scanf("%100s", name);

    int idx = 0;
    while(name[idx] != '\0') {
        if (!((name[idx] >= 'a' && name[idx] <= 'z') ||
            (name[idx] >= 'A' && name[idx] <= 'Z'))) {
            printf("The supplied name contains non-alphabetic symbols!\n");
        }
        return 1;
    }
    idx++;
}
```

Obsahujú zadané stringy iba písmená?

```
printf("Please specify your surname: ");
scanf("%100s", surname);
idx = 0;
while(surname[idx] != '\0') {
    if (!((surname[idx] >= 'a' && surname[idx] <= 'z') ||
        (surname[idx] >= 'A' && surname[idx] <= 'Z'))) {
        printf("The supplied surname contains non-alphabetic symbols
!\n");
        return 1;
    }
    idx++;
}

printf("Your full name is: %s %s\n", name, surname);

return 0;
}
```


Obsahujú zadané stringy iba písmená?

- 1 Načítajte zo STDIN 2 stringy, ktorých dĺžka je maximálne 100 znakov
- 2 Pre každý string vypíšte, či obsahuje iba písmená
- 3 Pre overenie správnosti využite vami vytvorenú funkciu

Obsahujú zadané stringy iba písmená?

```
#include <stdio.h>

int is_alpha(char c) {
    return (c >= 'a' && c <= 'z') || (c >= 'A' && c <= 'Z');
}

int is_name(char arr[]) {
    int idx = 0;
    while(arr[idx] != '\0') {
        if (!is_alpha(arr[idx])) {
            printf("The supplied surname contains non-alphabetic symbols\n");
            return 0;
        }
        idx++;
    }
    return 1;
}
```

Obsahujú zadané stringy iba písmená?

```
int main() {  
    char name(101);  
    char surname(101);  
  
    printf("Please fill in your name: ");  
    scanf("%100s", name);  
    if (!is_name(name)) {  
        return 1;  
    }  
  
    printf("Please fill in your surname: ");  
    scanf("%100s", surname);  
    if (!is_name(surname)) {  
        return 1;  
    }  
  
    printf("Your full name is: %s %s\n", name, surname);  
    return 0;  
}
```

Koľko krát obsahuje zadané slovo písmeno 'a'?

- 1 Načítajte zo STDIN string s maximálnou dĺžkou 100 znakov
- 2 Napíšte funkciu, ktorá pre daný string vypíše počet znakov 'a', ktoré sa v ňom nachádzajú

```
int main() {  
    char word(101);  
    int len = 0;  
  
    printf("Please fill in your text: ");  
    scanf("%100s", word);  
    // There will be function call  
  
    printf("Your text contains %d 'a' letters\n", len);  
    return 0;  
}
```

Koľko krát obsahuje zadané slovo písmeno 'a'?

```
#include <stdio.h>

int a_count(char word[]) {
    int i = 0, cout = 0;

    while (word[i] != '\\0') {
        if (word[i] == 'a') {
            cout += 1;
        }
        i ++;
    }

    return cout;
}

len = a_count(word);
```

Kde sa nachádza maximum v poli?

- 1 Načítajte zo STDIN do poľa o veľkosti 5 celé čísla.
- 2 Napíšte funkciu, ktorá pre dané pole vráti index najväčšieho čísla

```
int main() {  
    int arr_size = 5;  
    int arr1 ( arr_size );  
  
    printf("Please select five numbers: ");  
    for (int i = 0; i < arr_size; i++) {  
        scanf("%d", &arr1(i));  
    }  
  
    // There will be function call  
    printf("Maximum arr value is on index: %d\n", arr_max);  
  
    return 0;  
}
```

Kde sa nachádza maximum v poli?

```
#include <stdio.h>

int get_max(int arr[], int len) {
    int max_tmp = arr[0], indx = 0;
    for (int i = 1; i < len; i++) {
        if (arr[i] > max_tmp) {
            max_tmp = arr[i];
            indx = i;
        }
    }

    return indx;
}

int arr_max = get_max(arr1, arr_size);
```

Koľko čísiel sa opakuje v poli?

- 1 Načítajte dvakrát zo STDIN do poľa o veľkosti 5 celé čísla.
- 2 Napíšte funkciu, ktorá pre dané polia vráti počet spoločných prvkov

```
int main(void) {  
    int arr1(5);  
    int arr2(5);  
  
    printf("Select %d numbers for the first array: ", 5);  
    for (int i = 0; i < 5; i++) {  
        scanf("%d", &arr1(i));  
    }  
    printf("Select %d numbers for the second array: ", 5);  
    for (int i = 0; i < 5; i++) {  
        scanf("%d", &arr2(i));  
    }  
  
    int same_numbers = 0;  
    // How to solve this problem ?  
    printf("Number of same values: %d\\n", same_numbers);  
    return 0;  
}
```


Koľko čísiel sa opakuje v poli?

```
#include <stdio.h>
```

```
int is_in_array (int arr (), int len, int value) {  
    for (int i = 0; i < len; i++) {  
        if (arr(i) == value) {  
            return 1;  
        }  
    }  
    return 0;  
}  
  
for (int i = 0; i < 5; i++) {  
    if (is_in_array (arr2, 5, arr1(i))) {  
        same_numbers++;  
    }  
}
```

Stačí vypracovať jednu variantu

- Implementujte 2 funkcie, kde jedna vypíše slovo hodina a druhá slovo minúta v správnom tvare podľa zadaneho počtu hodín, alebo minút. Príklad: pre vstup 5 prvá funkcia vypíše string 5 hodín, pre vstup 40 vypíše druhá funkcia string 40 minút.
- Načítajte zo vstupu dve 5 prvkové postupnosti čísiel do poľa. Následne spočítajte súčet prvkov v oboch poliach a vypíšte, ktorý je väčší. Pri riešení vytvorte a použite aspoň jednu funkciu.
- Načítajte zo vstupu jednu 5 prvkovú postupnosť čísiel do poľa. Určite, či sa jedná o množinu, teda či sú všetky čísla v poli unikátne. Pri riešení vytvorte a použite aspoň jednu funkciu.