

Algoritmy a programování

I

Mgr. Jan Schreier

- Kontakt: jan.schreier@slu.cz
- Kancelář 227
- Konzultační hodiny: dle domluvy

Obsah

- Základní konstrukce C++

Zápočet

- Zápočtový projekt
- Požadavky:
- V jazyce C++

Studijní materiály

- Prezentace, příklady ze cvičení naleznete v ISSU.

Otázky

Syntaxe

- Pokud soubor se zdrojovým kódem uložíte s příponou .C, měl by být jako poslední příkaz funkce main()

```
return 0;
```

Pokud soubor uložíte s příponou .CPP, tak to obvykle není nutné

Syntaxe

```
#include <iostream>
using namespace std;

int main() {
    cout << "Hello World!";
    return 0;
}
```

- Načtení knihovny pomocí direktivy #include
- Abychom nemuseli psát stále odkaz na prostor jmen, lze dosáhnout pomocí direktivy using, std::cout << "Hello World"
- Main() je hlavní funkce programu a její kód píšeme do složených závorek { },

Komentáře

- `//` Toto je jednořádkový komentář: od dvojice lomítek do konce řádku.
- `/*` Toto je víceřádkový komentář `*/`

Datové typy

Datový typ	Co s ním	V printf/scanf:	Příklad:
int	celé číslo, délka 4 B	%d (desítkově) %x, %X (hex)	int i = 25;
short	celé číslo, délka 2 B		short s = 25;
long	celé číslo, délka 8 B		long x = 25;
float	racionální číslo, délka 4 B	%f (des. tečka) %e (exp. tvar)	float m = 3.82;
double	racionální číslo, délka 8 B	%f	double d = 82.55;
long double	racionální číslo, délka 12 B	%Lf (velé „L“!)	
char	znak, délka 1 B, vnitřně je to taky číslo	%c	char c = 'a';
unsigned int	celé nezáporné číslo, délka 4 B	%u	
unsigned float	racionální nezáporné číslo, délka 4 B		
string	řetězec (proměnné s knihovnou)	%s	
void	tam, kde přímo nechceme žádný datový typ psát (= neurčitý typ)		

Datové typy

```
int rok;                // deklarace bez inicializace
int rok = 2022;        // deklarace s inicializací
int den, mesic, rok = 2022; // deklarace tří proměnných, jedna je inicializována
```

Datové typy

- Kromě proměnných můžeme používat konstanty.
- Konstanta bývá taky určitého datového typu, ale na rozdíl od proměnné se její hodnota nemění.

```
const double PI = 3.14159; // konstanta typu double
```

Výstup

```
#include <iostream>
using namespace std;

int main() {
    cout << "Hello World!";
    cout << "I am learning C++";
    return 0;
}
```

Vstupy a výstupy

- Pro práci se vstupy a výstupy je nutný knihovna string
- `#include <string>`

Výstup

Escape sekvence	Význam	Escape sekvence	Význam
<code>\n</code>	symbol konce řádku	<code>\r</code>	návrat na začátek řádku
<code>\t</code>	tabulátor	<code>\v</code>	vertikální tabulátor

Vstup

```
int x;  
cout << "Type a number: "; // Type a number and press enter  
cin >> x; // Get user input from the keyboard  
cout << "Your number is: " << x; // Display the input value
```

Aritmetické operace

OPERÁTOR	VÝZNAM
+	sčítání
-	odčítání
*	násobení
/	dělení
%	modulo (zbytek po dělení)

Aritmetické operace

Zkrácený zápis

- $i+=5$ // inkrementace o 5
- $i-=5$
- $i*=5$

Inkrement

- $X++$
- $X--$

Cvičení

- Získejte vstup od uživatele a ten zdvojnásobte

Relační operátory

OPERÁTOR	VÝZNAM
>	větší než
<	menší než
>=	větší nebo rovno
<=	menší nebo rovno
==	je rovno
!=	není rovno

Logické operátory

OPERÁTOR	VÝZNAM
&&	spojka a (and)
	spojka nebo (or)
!	negace

Příkaz IF

```
if (podmínka)
    příkaz, když je podmínka splněna
```

```
if (podmínka)
    příkaz, když je podmínka splněna
else
    příkaz, když podmínka není splněna
```

Příkaz IF

```
#include <iostream>

using namespace std;

int main() {
    int cislo = 10;

    if (cislo > 5) {
        cout << "Cislo je vetsi nez 5.";
    } else {
        cout << "Cislo neni vetsi nez 5.";
    }

    return 0;
}
```

Příkaz IF

```
if (condition1) {  
    // block of code to be executed if condition1 is true  
} else if (condition2) {  
    // block of code to be executed if the condition1 is false and condition2 is true  
} else {  
    // block of code to be executed if the condition1 is false and condition2 is false  
}
```

Příkaz IF

```
#include <iostream>

using namespace std;

int main() {
    int cislo = 19;

    if (cislo > 15) {
        cout << "Cislo je vetsi nez 15.";
    } else if (cislo > 5) {
        cout << "Cislo je vetsi nez 5.";
    } else {
        cout << "Cislo neni vetsi nez 5.";
    }

    return 0;
}
```

Cvičení

- Validátor věku
- Získejte od uživatele jeho věk a podle něj vypište, zda je plnoletý

Cvičení

```
#include <iostream>

using namespace std;

int main() {
    int vek;
    int minimalniVek = 18;

    cout << "Zadejte vek: ";
    cin >> vek;

    if (vek >= minimalniVek) {
        cout << "Osoba je starší než " << minimalniVek << " let." << endl;
    } else {
        cout << "Osoba je mladší než " << minimalniVek << " let." << endl;
    }

    return 0;
}
```

Switch

```
switch (výraz) {  
  case hodnota1:  
    příkazy;  
    break;  
  case hodnota2:  
    příkazy;  
    break;  
  case hodnota3:  
    příkazy;  
    break;  
  ...  
  default:  
    příkazy;  
}
```

Switch

```
int main() {
    int den;

    cout << "Zadejte číslo dne v týdnu (1-7): ";
    cin >> den;

    switch (den) {
        case 1:
            cout << "Pondělí" << endl;
            break;
        case 2:
            cout << "Úterý" << endl;
            break;
        case 3:
            cout << "Středa" << endl;
            break;
        case 4:
            cout << "Čtvrtek" << endl;
            break;
        case 5:
            cout << "Pátek" << endl;
            break;
        case 6:
            cout << "Sobota" << endl;
            break;
        case 7:
            cout << "Neděle" << endl;
            break;
        default:
            cout << "Neplatný den v týdnu." << endl;
    }

    return 0;
}
```

Cvičení

- Kalkulačka pomocí switch (násobení, dělení, sčítání, odčítání)