

C++ v Linuxe

Tomáš Srna

InstallFest 2011

5. března 2011

Obsah

- 1 Kompilácia
 - hello.cpp
 - g++
 - Makefile
- 2 Základy
 - Premenné
 - Podmienky
 - Cykly
- 3 Funkcie
- 4 Pointery, Polia, objekty
 - Pointre
 - Polia
 - Triedy a objekty
- 5 C++ Akademie SH

Obsah

- 1 Kompilácia
 - hello.cpp
 - g++
 - Makefile
- 2 Základy
 - Premenné
 - Podmienky
 - Cykly
- 3 Funkcie
- 4 Pointery, Polia, objekty
 - Pointre
 - Polia
 - Triedy a objekty
- 5 C++ Akademie SH

Hello World!

hello.cpp

```
#include <iostream>
using namespace std;
int main(int argc, char** argv)
{
    char hello[] = "Hello World!";
    char ahoy[] = "Ahoj svet!";
    char* napis = NULL;
    napis = hello;
    napis = ahoy;
    cout << napis << endl;
    return 0;
}
```

Kompilácia

```
g++
```

```
g++ -Wall -ggdb -pedantic -ohello hello.cpp
```

Hello World!

Makefile

```
all: hello
```

```
hello:
```

```
    g++ -Wall -ggdb -pedantic hello.cpp -ohello
```

```
clean:
```

```
    rm hello
```

Pozor! Pod xyz: musí príkaz začínať tabulátorom. Aj prázdny.

make

```
make all; make hello; make clean
```

```
make
```

make bez parametra "make" prvý cieľ.

Obsah

- 1 Kompilácia
 - hello.cpp
 - g++
 - Makefile
- 2 Základy
 - Premenné
 - Podmienky
 - Cykly
- 3 Funkcie
- 4 Pointery, Polia, objekty
 - Pointre
 - Polia
 - Triedy a objekty
- 5 C++ Akademie SH

Celé čísla

Nemajú desatinnú časť.

Všetky existuju aj v neznamienkovej - unsigned verzii

Ich typy

char: má veľkosť 8 bitov

short: minimálne 16 bitov

int: minimálne tak veľké ako short

long: minimálne tak veľké ako int, a minimálne 32 bitov

Pozor

Pri delení týchto typov, dochádza k celočíselnému deleniu.

Čísla s pohyblivou desatinnou čiarkou

Ich typy

float

double

long double

bool

Typ pre logické hodnoty. 0 je false, všetko iné je true.

char

Znak. Môže mať 256 hodnôt - reprezentuje znak.

Podmienka if-else

Syntax

```
if (podmienka)
    telo bloku
else if (podmienka)
    telo bloku
else if (podmienka)
    telo bloku
else
    telo bloku
```

Podmienka if-else

Príklad:

```
#include <iostream>
using namespace std;
int main(void){
a=5;
if (a==5) cout << "Cislo sa rovna 5" <<endl;
else if (a==6)
{cout << "Cislo sa rovna 6"; cout<<endl;}
else if (a<5) cout << "Cislo mensie nez 5" <<endl;
else cout << "Cislo vacsie nez 6" <<endl;
return 0;
}
```

Prepínač switch

Syntax:

```
switch (vyraz)
{
    case hodnota:
        prikazy;
        break;
    case hodnota:
        prikazy;
        break;
    default:
        prikazy;
        break;
}
```

Prepínač switch

Príklad:

```
#include <iostream>
using namespace std;
int main(void){
cout<<"Zadaj A alebo B. "<<endl;
char c;
cin>>c;
switch (c)
{
    case 'a':
    case 'A': cout<<"Zadal si A."<<endl; break;
    case 'b':
    case 'B': cout<<"Zadal si B."<<endl; break;
    default : cout<<"Nauc sa citat."<<endl; break;
}
```

Podmienený výraz (ternárny operátor ?:)

Syntax:

```
vyraz ? b : c
```

Príklad:

Ošetrenie delenia nulou:

```
vysledok = i ? 128 / i : 0;
```

Cyklus for

Syntax:

```
for (inicializacia; podmienka; vyraz)
    telo cyklu
```

- Inicializace môže obsahovať deklaráciu
- V podmienke treba dávať pozor na operátory = a ==

Pozor!

Keď chcete dať viac než jeden príkaz do cyklu, treba použiť lomené zátvorky.

Cyklus for

Príklad:

```
#include <iostream>
using namespace std;
int main(void){
cout<<"Zadaj pocet cyklov. "<<endl;
int pocet;
cin>>pocet;
for(int i=0;i<pocet;i++){
cout<<"Hodnota i je: "<<i;
cout<<endl;
}
return 0;
}
```


Obsah

- 1 Kompilácia
 - hello.cpp
 - g++
 - Makefile
- 2 Základy
 - Premenné
 - Podmienky
 - Cykly
- 3 Funkcie
- 4 Pointery, Polia, objekty
 - Pointre
 - Polia
 - Triedy a objekty
- 5 C++ Akademie SH

Funkcie

Funkcie sú základným stavebným kameňom jazyka C++.

Definícia funkcie

```
navratovy_typ nazov ([parametre])  
{  
    telo funkcie  
}
```

- Názov funkcie slúži na jej identifikáciu.
- Parametre funkcie sú očakávané dáta.
- Funkcia má vždy jednu návratovú hodnotu. Pokiaľ ich chceme viac, použijeme tzv. výstupné parametre (pointre alebo referencie).

Obsah

- 1 Kompilácia
 - hello.cpp
 - g++
 - Makefile
- 2 Základy
 - Premenné
 - Podmienky
 - Cykly
- 3 Funkcie
- 4 Pointery, Polia, objekty**
 - Pointre
 - Polia
 - Triedy a objekty
- 5 C++ Akademie SH

Pointre

Čo je to pointer?

Pointer je v podstate odkaz na pamäť.

Deklarácia

```
int* a;  
char* b;
```

Operácie s pointrami

* - dereference
& - adress of

Ukážeme si čo sme a čo nesme s pointrami.

(NULL pointer, mimo alokovanej pamäti, neinicializovaný pointer)

Polia

Alokácia

- Statická
- Dynamická

Statická alokácia

```
int a[5];
```

Dynamická alokácia

```
int* a = new int[5];
```

Čo s viacrozmernými poliami?

Viacrozmerne polia

Statická alokácia

```
int a[5][25];
```

Dynamická alokácia

```
int i;  
int* a = new int*[5];  
for (i=0; i<5; ++i)  
    a[i] = new int[25];
```

Triedy

Základná trieda

```
class CRectangle {  
    int x, y;  
public:  
    void set_values (int,int);  
    int area (void);  
};
```

Objekty

Objekt je instanciou triedy.

```
CRectangle rect;
```

Obsah

- 1 Kompilácia
 - hello.cpp
 - g++
 - Makefile
- 2 Základy
 - Premenné
 - Podmienky
 - Cykly
- 3 Funkcie
- 4 Pointery, Polia, objekty
 - Pointre
 - Polia
 - Triedy a objekty
- 5 C++ Akademie SH

C++ Akademie SH

- Pokiaľ ťa programovanie v C++ zaujalo, zapíš sa na C++ Akademiu SH
- Info na <http://c.siliconhill.cz>