



FAKULTA ELEKTROTECHNIKY **ústav**
A KOMUNIKAČNÍCH **teoretické a experimentální**
TECHNOLOGIÍ **elektrotechniky**

Seminář C++

1. přednáška

Autor: doc. Ing. Jan Mikulka, Ph.D.

2023



Obsah přednášky

- Organizace výuky
- Rozsah kurzu
- Vývojové prostředí
- Základy strukturovaného programování v jazyce C, datové typy, funkce, podmínky, cykly
- Příklady k řešení

Organizace výuky

- Garant
 - doc. Ing. Jan Mikulka, Ph.D.
- Vyučující
 - doc. Ing. Jan Mikulka, Ph.D. (mikulka@vut.cz, SC4.87)
 - konzultace: v konzultačních hodinách a po domluvě e-mailem

Organizace výuky

- 1 x 3 hod. týdně
 - 1 hod. přednáška
 - 2 hod. počítačové cvičení

Hodnocení

- Klasifikovaný zápočet
- Během výuky
 - **40 bodů** za průběžně vypracovaná zadání a projekty malého rozsahu
- Na konci semestru
 - **60 bodů** za semestrální projekt
 - zadání a výběr projektu v půlce semestru

Rozsah kurzu

1. Základy strukturovaného programování v jazyce C, datové typy, funkce, podmínky, cykly.
2. Dynamická alokace paměti, vícerozměrná pole, pointery, vracení hodnoty parametrem.
3. Vlastnosti jazyka C/C++, směrnice ANSI, objektově orientované programování, algoritmy.
4. Klíčová slova C++, třídy, členské proměnné.
5. Členské funkce-metody, ukazatel this, agregáty, oblasti platnosti a dosažitelnosti identifikátorů.
6. Prostory jmen, paměťové třídy, objekty.

C

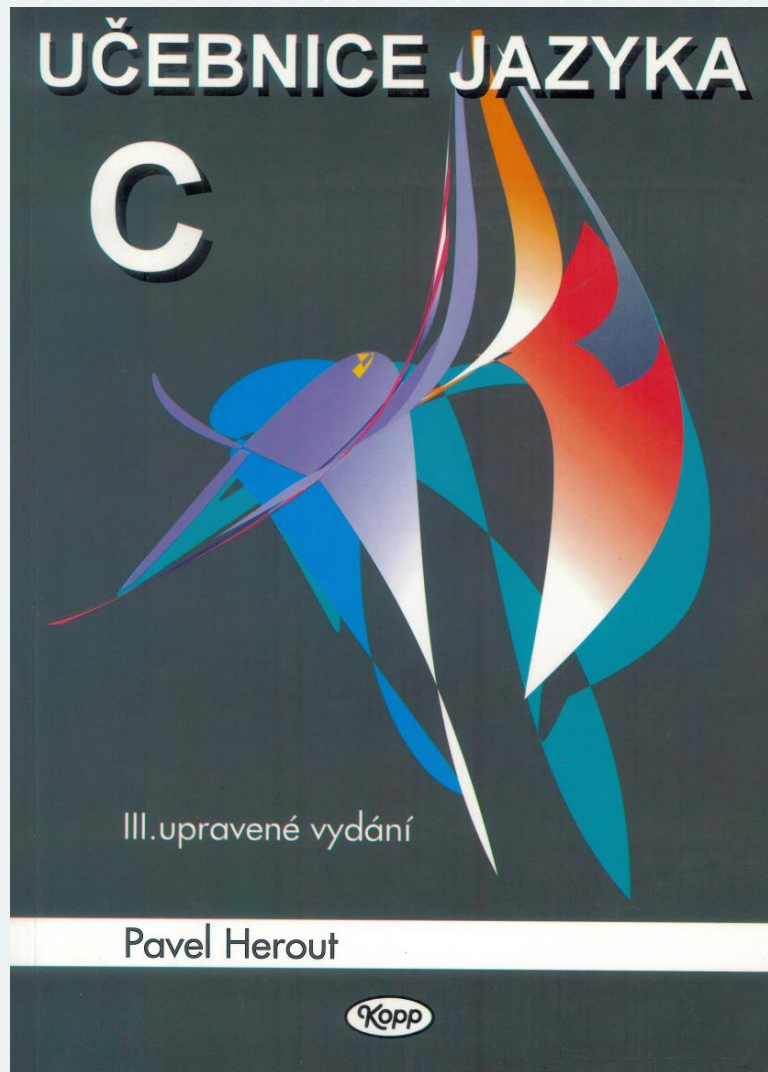
C++

Rozsah kurzu

7. Přiřazování objektů, předávání a vracení objektů funkcím.
8. Přehled operátorů, popis a užití operátorů.
9. Přetěžování operátorů.
10. Dědičnost, přístupy v dědičnosti, polymorfismus.
11. Přetěžování konstruktorů, anachronismus přetěžování, přetěžování a nejednoznačnost.
12. Šablony funkcí a tříd, parametry šablon, šablony objektových typů, obsluha výjimek.
13. Knihovny STL a jejich aplikace.

C++

Doporučená literatura C



Pokud to myslíte s jazykem C opravdu vážně, pak by vám *Učebnice jazyka C* neměla v knihovně chybět!

Výhodou této knížky je, že je to opravdu učebnice a ne manuál. Krůček po krůčku vás povede úskalími jazyka C a nabídne vám postupně stále širší pohled na možnosti, které vám jazyk C může poskytnout.

Kromě podrobného výkladu všech rysů jazyka C a množství příkladů se zde mimo jiné dočtete i o nejčastějších chybách, které dělají začínající programátoři, naleznete tu návody, jak psát přehledné programy nebo jak spolupracovat na velkém programu se svými kolegy bez rizika zbytečných nepříjemností.

Jestliže si sami naprogramujete všech asi 120 příkladů ze cvičení, pak můžete o sobě na konci této učebnice prohlásit, že jazyk C ovládáte na velmi dobré úrovni.

Ke knížce si můžete dokoupit ještě disketu s asi 250 kompletními odladěnými programy. Jsou zde zdrojové texty všech příkladů z knihy a všech vyřešených cvičení.

Autor (* 1961) pracuje jako odborný asistent na katedře informatiky a výpočetní techniky Západočeské univerzity v Plzni. Zabývá se výukou softwareově orientovaných předmětů. Značné zkušenosti s využíváním jazyka C získal jako spoluvůdce dvou velkých projektů — z oblasti komunikací a z oblasti simulací. Je autorem nebo spoluautorem dvou skript a pěti odborných knih, jejichž obliba je přičítána i tomu, že je v nich uvedeno něco víc, než bývá v manuálech — osobní praktické zkušenosti.

ISBN 80-85828-21-9

Nakladatelství KOPP

Cena knihy včetně DPH 149,- Kč
Cena diskety včetně DPH 69,- Kč



9 788085 828214 >

Doporučená literatura C



Kniha velmi těsně navazuje na **Učebnici jazyka C** od stejného autora a stejného nakladatelství. Obsahuje řadu informací, návodů a zkušeností, které ocení nejspíše programátoři, kteří jazyk C používají již víceméně suverénně.

Stěžejní části knihy tvoří detailní rozbor všech možných činností se soubory a pak také popisy standardních knihoven. Proto je vhodná pro ty, programátory, kteří občas narazí na něco, co by potřebovali vědět, ne jen tušit, případně nemají čas bádát nad něčím, co je již prozkoumáno a popsáno.

Ke knížce si můžete zdarma stáhnout asi 350 kompletních odladěných programů. Jedná se o zdrojové soubory všech příkladů z knihy.

Autor (*1961) pracuje na katedře informatiky a výpočetní techniky Západočeské univerzity v Plzni. Zabývá se výukou softwarově orientovaných předmětů. Značné praktické zkušenosti s jazykem C získal jako spoluvůdce tří velkých projektů. Je autorem nebo spoluautorem několika skript a řady odborných knih, jejichž obliba je připočítána i tomu, že v nich uvedeno něco víc, než bývá v manuálech – osobní praktické zkušenosti.



nakladatelství

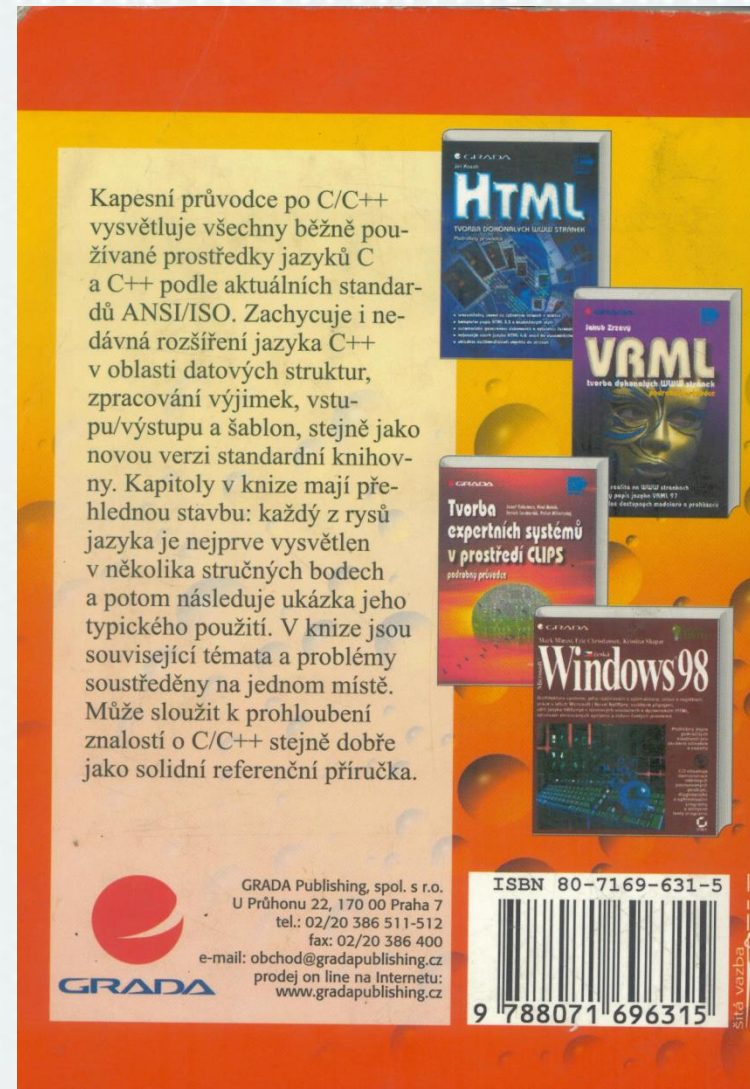
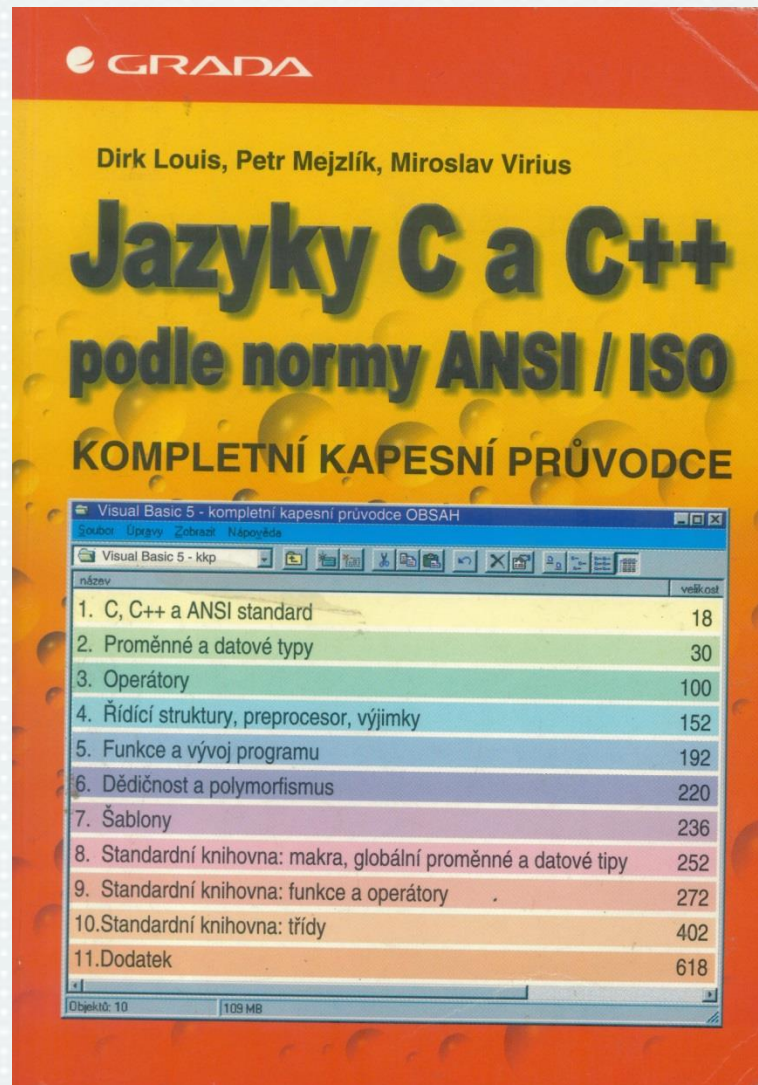
Šumavská 3, 370 01 České Budějovice
Tel./fax: 386 460 474
e-mail: knihy@kopp.cz; Internet: www.kopp.cz
Cena knihy 149,- Kč včetně DPH

ISBN 80-7232-221-4



9 788072 322213

Doporučená literatura C++

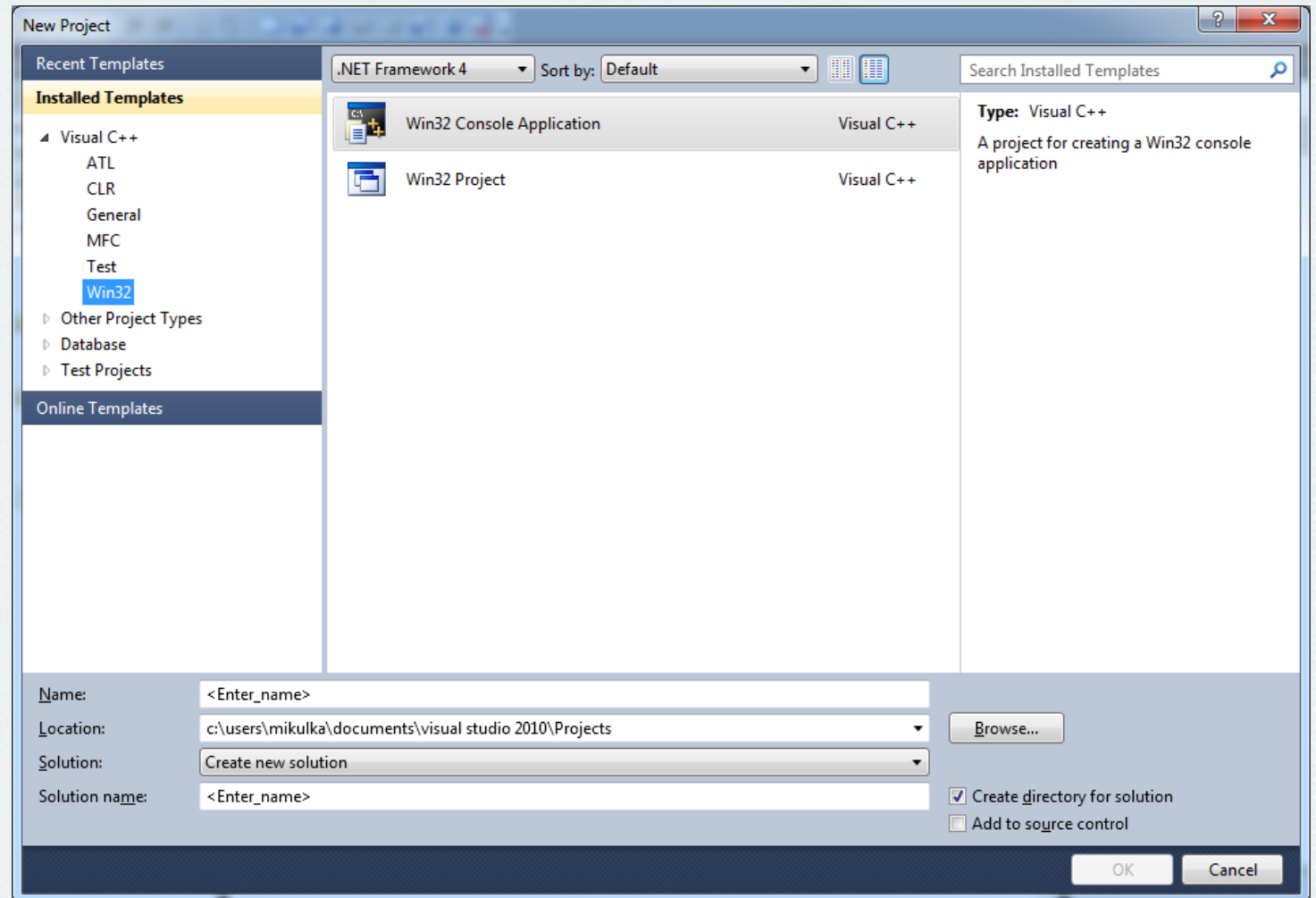


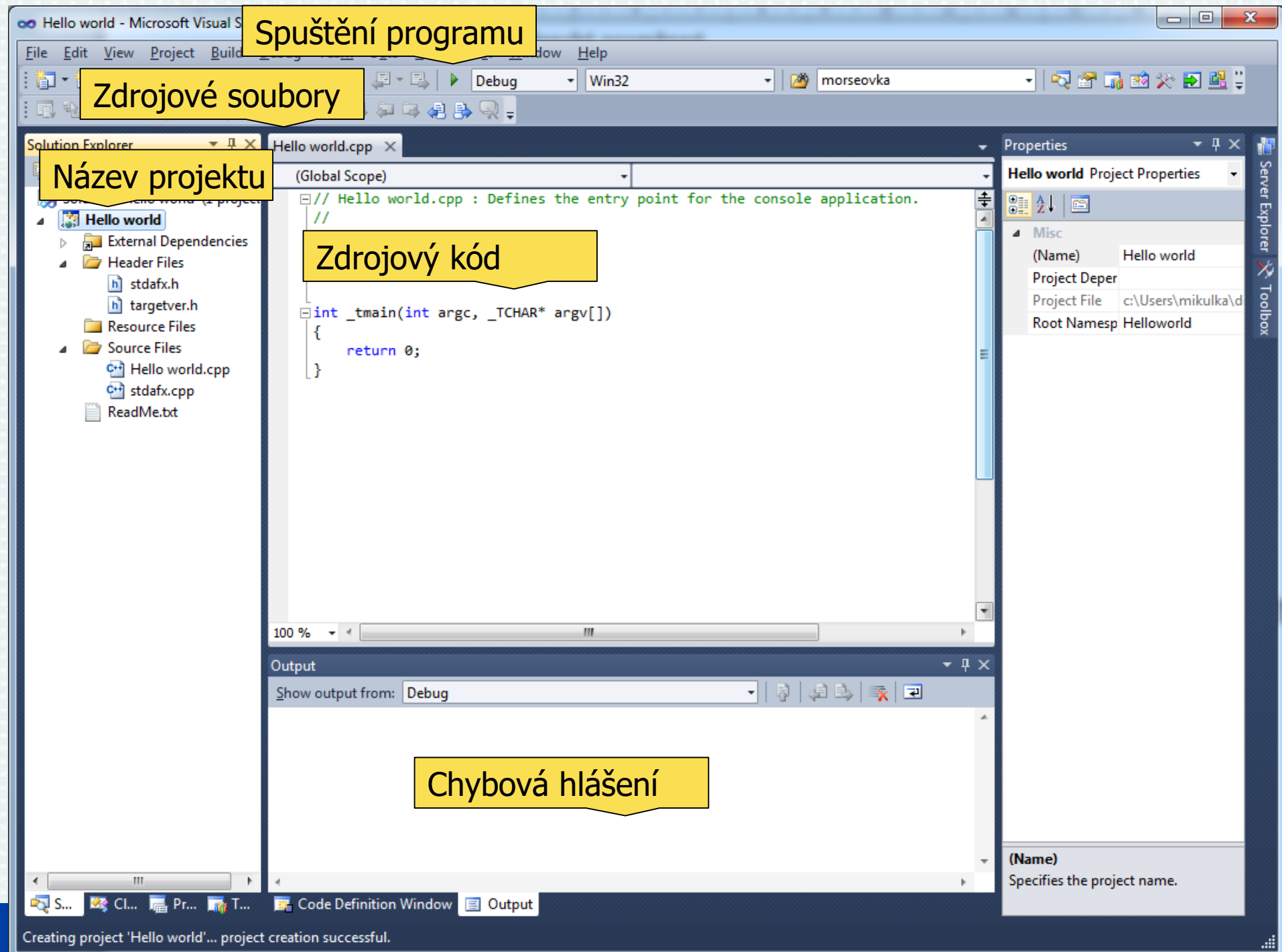
Vývojové prostředí

- Visual Studio 20xx
 - (MSDNAA licence pro studenty VUT zdarma)
 - <https://msdnaa.feec.vutbr.cz/web/>
- Visual Studio Community Edition
 - (licence zdarma i pro komerční účely)
 - <https://www.visualstudio.com/cs/vs/community/>
- Pro účely procvičování C++ budeme používat konzolové aplikace (Win32 Console Application)
- V případě potřeby CLR/Windows Form Application

Vývojové prostředí

- File -> New -> Project...



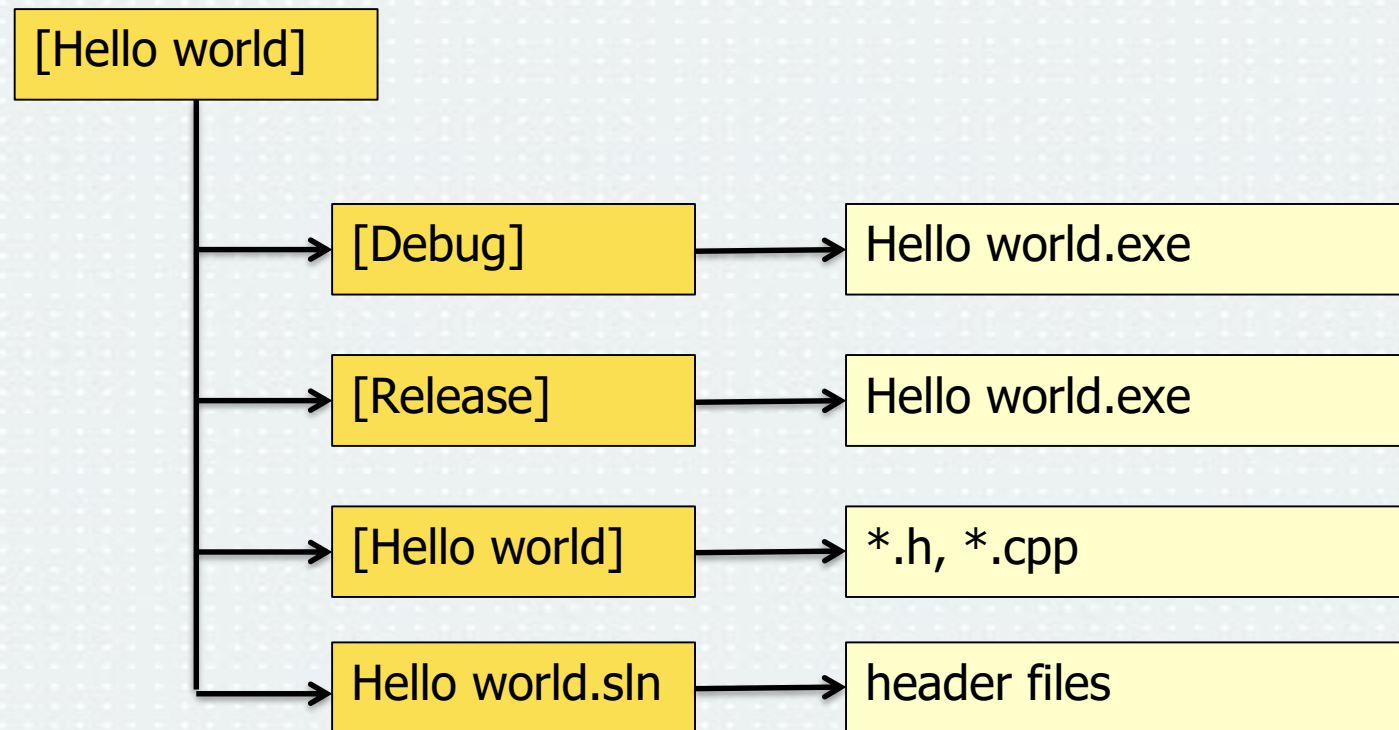


Vývojové prostředí

- Důležité klávesové zkratky VS
 - F5: spuštění programu
 - F7: kompilace projektu
 - F9: vloží/odstraní breakpoint
- Při běhu programu
 - Ctrl + C: ukončení programu

Vývojové prostředí

- Adresářová struktura projektu VS



Základy strukturovaného programování

- Základem strukturovaného programování (SP) je dělení problému na dílčí podproblémy a jejich implementace pomocí **funkcí**.
- Na rozdíl od objektově orientovaného programování (OOP) není možné psát univerzální hierarchicky uspořádaný kód.
- Kód SP se tak stává do jisté míry nereprodukovatelným.

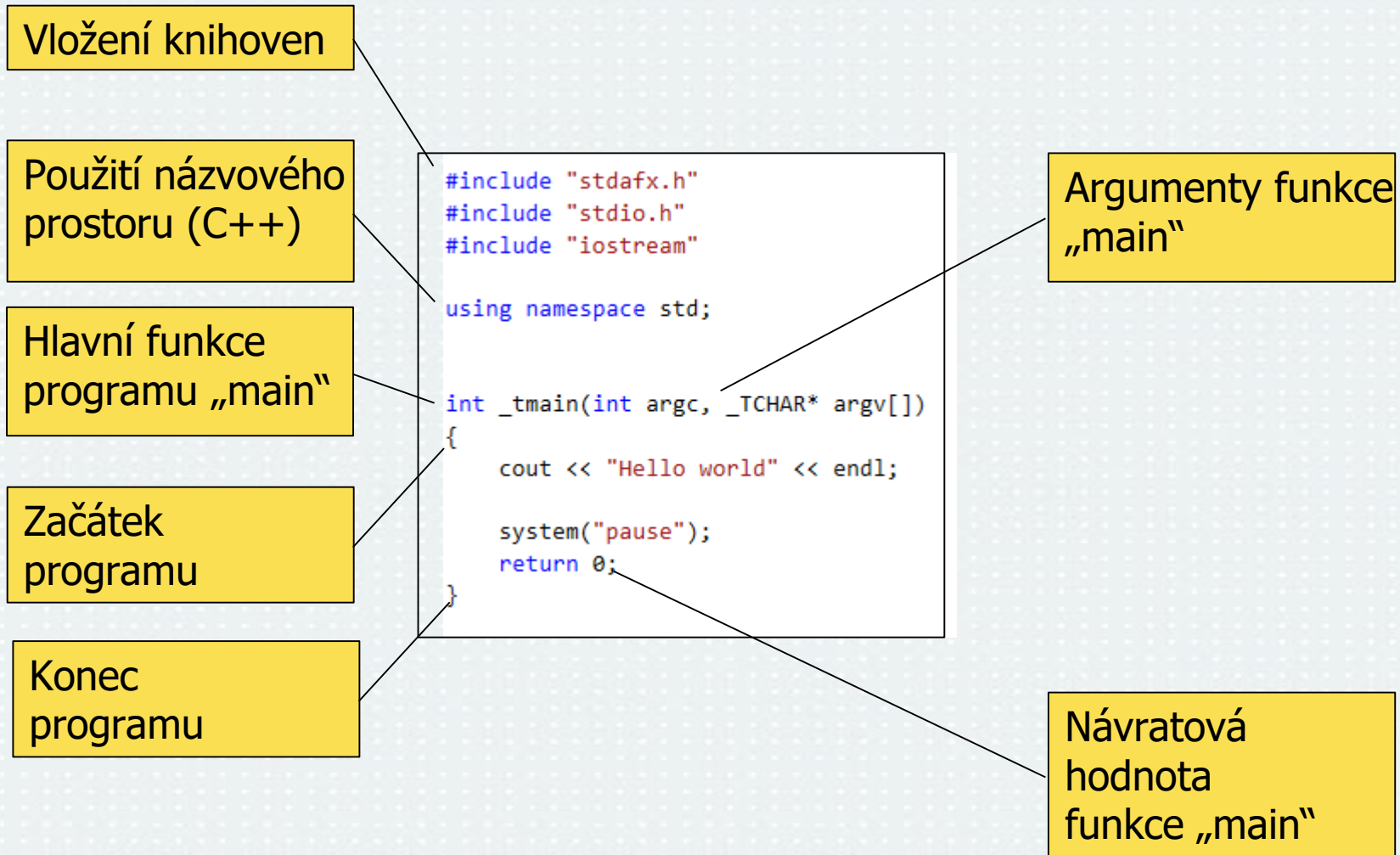
Základy jazyka C

- Na co se zaměříme?
 - Typy datových proměnných.
 - Funkce.
 - Přiřazení, konstanty.
 - Podmínky, cykly.

Datové typy jazyka C

Datový typ	Počet bitů	Význam
char	8	Znak
short	16	Krátké celé číslo
int	16 nebo 32	Celé číslo
long	32	Dlouhé celé číslo
float	32	Racionální číslo
double	64	Racionální číslo
long double	80	Racionální číslo

Struktura programu



Hlavní funkce „main“

- Po spuštění programu je vždy volána funkce s názvem „main“.
- Syntaxe funkce „main“

```
int main(int argc, char* argv[])
```

Typ návratové
hodnoty

Typ parametru
argc

Typ parametru
argv[]

Hlavní funkce „main“

```
int main(int argc, char* argv[])
```

- Parametr „argc“
 - Udává počet parametrů, které spuštěný program dostal v příkazové řádce
- Parametr „argv[]“
 - Jde o dvourozměrné pole, ve kterém můžeme nalézt argc parametrů zadaných v příkazové řádce

Příklad na použití parametrů v příkazové řádce

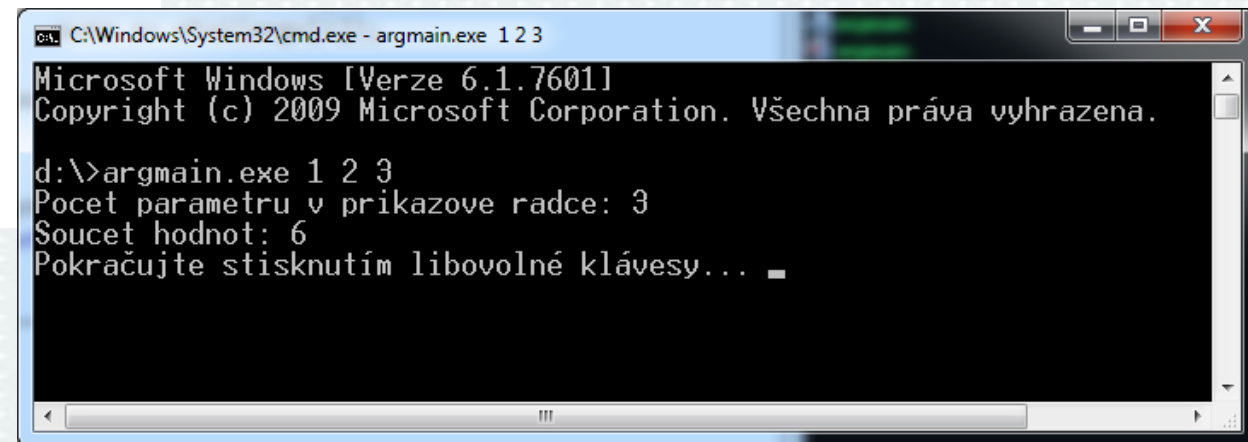
```
int main(int argc, char* argv[])
{
    int suma = 0;

    cout << "Pocet parametru v prikazove radce: " << argc-1 << endl;
    cout << "Soucet hodnot: ";

    for (int i = 1; i < argc; i++)
        suma += atoi(argv[i]);

    cout << suma << endl;

    system("pause");
    return 0;
}
```

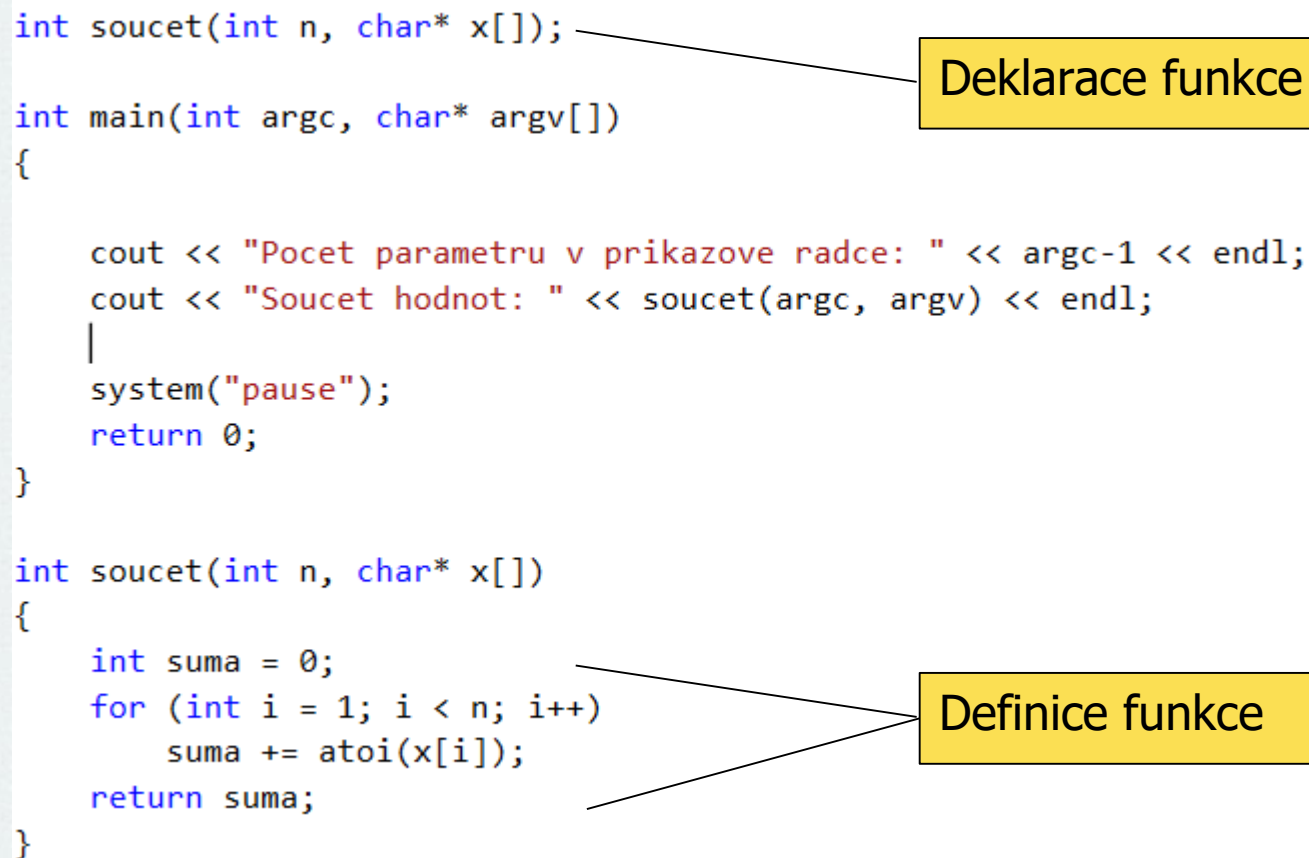


The screenshot shows a Windows command prompt window titled "C:\Windows\System32\cmd.exe - argmain.exe 1 2 3". The window displays the output of the program: "Microsoft Windows [Verze 6.1.7601] Copyright (c) 2009 Microsoft Corporation. Všechna práva vyhrazena. d:\>argmain.exe 1 2 3 Pocet parametru v prikazove radce: 3 Soucet hodnot: 6 Pokračujte stisknutím libovolné klávesy...". The user has entered the command "argmain.exe 1 2 3" and the program has executed successfully, displaying the number of parameters (3) and their sum (6).

Definice/deklarace funkce v C

- Strukturované programování = postupné volání funkcí v hlavním programu (ve funkci „main“).

```
int soucet(int n, char* x[]);  
  
int main(int argc, char* argv[])  
{  
  
    cout << "Pocet parametru v prikazove radce: " << argc-1 << endl;  
    cout << "Soucet hodnot: " << soucet(argc, argv) << endl;  
    |  
    system("pause");  
    return 0;  
}  
  
int soucet(int n, char* x[])  
{  
    int suma = 0;  
    for (int i = 1; i < n; i++)  
        suma += atoi(x[i]);  
    return suma;  
}
```



Deklarace funkce

Definice funkce

Funkce v C

- V případě, že funkce nevrací návratovou hodnotu (procedura), případně nemá žádné parametry, použijeme typ **void**

```
void print_head(void)
{
    cout << "Vítejte v programu pro scitání!";
}

int main(int argc, char* argv[])
{
    print_head();
    cout << "Pocet parametru v prikazove radce: " << argc-1 << endl;
    cout << "Soucet hodnot: " << soucet(argc, argv) << endl;

    system("pause");
    return 0;
}
```


Konstanty v C

```
int main(int argc, char* argv[])
{
    const int c = 2;

    cout << c;

    c = 3;

    cout << c;

    system("pause");
    return 0;
}
```

```
#define c 2

int main(int argc, char* argv[])
{
    cout << c;

    c = 3;

    cout << c;

    system("pause");
    return 0;
}
```

error C2106: '=' : left operand must be l-value

error C3892: 'c' : you cannot assign to a variable that is const

Podmínky v C

- Klíčová slova **if**, **else if**, **else**.

```
int main(int argc, char* argv[])
{
    cout << "Pocet parametru v prikazove radce: " << argc-1 << endl;
    if (argc-1 == 2)
        cout << "Soucet hodnot: " << atoi(argv[1]) + atoi(argv[2]) << endl;
    else if (argc-1 <=1)
        cout << "Prilis malo parametru!" << endl;
    else
        cout << "Prilis mnoho parametru!" << endl;

    system("pause");
    return 0;
}
```

Cykly v C

- V jazyce C lze použít tři typy cyklů:
 - **for**
 - **do – while**
 - **while**

Cyklus for

```
int soucet(int n, char* x[]);

int main(int argc, char* argv[])
{
    cout << "Pocet parametru v prikazove radce: " << argc-1 << endl;
    cout << "Soucet hodnot: " << soucet(argc, argv) << endl;
    |
    system("pause");
    return 0;
}

int soucet(int n, char* x[])
{
    int suma = 0;
    for (int i = 1; i < n; i++)
        suma += atoi(x[i]);
    return suma;
}
```

Počáteční inicializace
řídící proměnné

Podmínka ukončení cyklu

Přiřazení nové hodnoty
řídící proměnné

Cyklus do - while

```
int soucet(int n, char* x[]);

int main(int argc, char* argv[])
{
    cout << "Pocet parametru v prikazove radce: " << argc-1 << endl;
    cout << "Soucet hodnot: " << soucet(argc, argv) << endl;

    system("pause");
    return 0;
}

int soucet(int n, char* x[])
{
    int suma = 0, c = 0;
    do
    {
        c++;
        suma += atoi(x[c]);
    } while (c < n-1);
    return suma;
}
```

Počáteční inicializace
řídící proměnné

Přiřazení nové hodnoty
řídící proměnné

Podmínka ukončení cyklu

Cyklus while

```
int soucet(int n, char* x[]);

int main(int argc, char* argv[])
{
    cout << "Pocet parametru v prikazove radce: " << argc-1 << endl;
    cout << "Soucet hodnot: " << soucet(argc, argv) << endl;

    system("pause");
    return 0;
}

int soucet(int n, char* x[])
{
    int suma = 0, c = 0;
    while (c < n-1)
    {
        c++;
        suma += atoi(x[c]);
    };
    return suma;
}
```

Počáteční inicializace
řídící proměnné

Podmínka ukončení cyklu

Přiřazení nové hodnoty
řídící proměnné

**Děkuji vám
za pozornost.**

mikulka@vut.cz
www.utee.fekt.vut.cz