

Programování v jazyce C IPJCE

Cíle

- číst a psát programy v jazyce C se znalostí základních knihoven
- porozumění pointerům, alokace a uvolnění paměti
- pochopení odděleného překladu, vytváření makefile

Předpokládané znalosti

- základy znalosti algoritmizace
- znalost pojmu proměnná, konstanta, metoda
- znalost řídicích struktur if/else, cyklů for, while, do-while
- základní znalost syntaxe "céčkových" jazyků (C, C++, C#, nebo Java...)

Literatura/Materiály

- Herout P.: Učebnice jazyka C (kterékoliv vydání)
- Přednášky a materiály ode mne, případně Jan Fikejz, Martin Hájek
- Internet <http://www.cprogramming.com/> (Aj)
http://en.wikibooks.org/wiki/C_Programming (Aj)
<http://programujte.com/clanky/43-serial-c/> (Cz)

Organizace předmětu

Přednášky - nepovinné avšak výrazně doporučené, na cvičení se předpokládá znalost z přednášky!

Cvičení - nepovinné, na každém cvičení je jeden bodovaný příklad, součet bodů ze cvičení je součástí hodnocení!

Hodnocení Předmětu

Předmět je zakončen klasifikovaným zápočtem

Známka se skládá z bodů ze cvičení, semestrální práce, zkouškového příkladu.

- Body ze cvičení se získávají za řešení typového příkladu na cvičení (nejjednodušší forma získání bodů, možná pomoc vyučujícího)
- Semestrální práce - samostatná práce odevzdávaná na konci semestru
- Zkouškový příklad
 - v případě dostatečného bodového plnění jednoduchý program na 10-15 min
 - v případě malého počtu bodů ze cvičení složitější program na cca 60 min

Uvědomte si, že programování v jazyce C je přes podobnost tomu co jste již dělali principiálně odlišné a může přes počáteční zdání jednoduchosti způsobit při podcenění značné potíže. Proto pokud čemukoliv nebudete rozumět, okamžitě se hlase.

Historie jazyka

- předchůdce = jazyk B
- 1972 – Dennis M. Ritchie, AT&T Bell Labs jazyk C (odstranění problémů jazyka B)
- masový úspěch – UNIX přepsán celý do C 1970 – 1978 K&R Standard („Pre-ANSI C“) shrnuto v [Kernighan, Ritchie]
- 1982 počátek standardizace jazyka a knihoven
- 1989 American National Standard X3.159 – 1989 = ANSI C“ (úprava jazyka dle praktických zkušeností)
- 1990 převzata pod ISO/IEC 9899-1990 nejnovější standard ISO/IEC 9899:1999 = „C99)

Výhody

- multiplatformní – „pro vše, co má CPU, existuje C kompilátor“ přenositelný – kód pro jednu platformu platný i pro jiné napříč OS, napříč CPU
- nízko úroňový – přímý přístup do paměti, bitové operace
- rychlý, vhodný pro systémové programování (vývoj OS, Linux napsán kompletně v C) standardizovaný

Nevýhody

- nebezpečný – slabá typová kontrola, nejednoznačná syntaxe, nepřehledný
- nízko úroňový

```
#include <stdio.h>
#define PO(o,t)\
(((o>64)&&(o<91))?(((t>96)&&(t<123))?(t-32):(t)):(((t>64)&&(t<91))?(t+32):(t)))

void main( ) {
    char *poo="poot",
    *Poo="pootpoot", o[9];int o,t,T,p;(t=p
    ((o= getc( stdin ))!= (EOF))if ((p== =0)||(*o='\0');while
    (o<91 )) || (( o> 96 ) &&(o< 123) 0 ) && (((o>64)&&(
    t!=8 )&&(0 [t]= o)&& (0[++ t] = '\0')
    ;else {if (t>7) {for (T = 0 ; T <=7; T++ )
    printf("%c", PO(*( O+T), *(Poo+ T))); printf
    ("%c", o);}else if (t>3){for (T =0;T<=
    3;T++) printf
    ("%c", PO(*(O
    +T),*( poo+T

    ) ) ; printf( "%c" , o ) ; } else printf ( "%s%c" , o , o ) ; ( t = 0 ) || (
    * 0 = '\0' ) ; ( o == 60 ) && ( ++p ) ; ( o == 62 ) && ( p!=0 ) && ( --p ) ; } }
```

Potřebné nástroje pro programování v C

- pro PC (Intel x86)
- Intel C compiler
- GCC (GNU Compiler Collection)
 - o kvalitní, free (GPL licence) nejen C/C++ kompilátor
 - o napříč OS (Windows, Linux, Unix, Solaris, ...)
 - o napříč CPU (x86, ARM, AVR, SPARC, ...)
 - o 100% C99 kompatibilní
- Microsoft C/C++ compiler
 - o C99 pouze částečně, ANSI C jen tak tak
- Kompilátor vs. IDE - IDE pouze frontend kompilátoru

32 slov

auto	double	int	struct
break	else	long	switch
case	enum	register	typedef
char	extern	return	union
const	float	short	unsigned
continue	for	signed	void
default	goto	sizeof	volatile
do	if	static	while