

Obsah druhého dílu

Pozor:

- Číslování knihy je společné pro první i druhý díl Učebnice jazyka C (podrobnosti viz v předmluvě). To tedy znamená, že druhý díl začíná kapitolou 17 a stranou 272. Všechny strany obsahu a předmluvy jsou číslovány římskými číslicemi.

| | |
|---|-----|
| Předmluva k aktuálnímu vydání | vi |
| 17 Komplexní pohled na souborový vstup a výstup | 272 |
| 17.1 Rozdíly mezi binárním a textovým souborem obecně | 273 |
| 17.1.1 Binární soubory | 273 |
| 17.1.2 Textové soubory | 274 |
| 17.2 Rozdíly mezi binárním a textovým režimem otevírání souboru | 274 |
| 17.3 Otevření souboru | 277 |
| 17.3.1 Různé způsoby otevírání souboru | 278 |
| 17.4 Uzavření souboru | 281 |
| 17.5 Ošetření chyb | 283 |
| 17.6 Čtení a zápis dat | 285 |
| 17.6.1 Formátované vstupy a výstupy | 286 |
| 17.6.2 Neformátované vstupy a výstupy | 286 |
| Zpracování jednoho znaku v jeden okamžik | 287 |
| Zpracování celé řádky najednou | 290 |
| Zpracování celého bloku dat najednou | 292 |
| 17.7 Bufferování | 293 |
| 17.7.1 Základní informace | 294 |
| 17.7.2 Možnosti bufferování poskytnuté ANSI C | 295 |
| 17.7.3 Funkce pro práci s bufferem | 295 |
| Funkce <code>setbuf()</code> | 295 |
| Funkce <code>setvbuf()</code> | 296 |
| 17.7.4 Řešení problému nebufferovaného <code>stdin</code> | 297 |
| 17.8 Funkce pro zápis bufferu na disk | 299 |
| 17.8.1 Funkce <code>fflush()</code> | 299 |

| | | |
|-----------|--|------------|
| 17.9 | Přímý přístup do souboru | 300 |
| 17.9.1 | Funkce <code>fseek()</code> – posun v souboru | 300 |
| 17.9.2 | Funkce <code>ftell()</code> – zjištění pozice v souboru | 300 |
| 17.9.3 | Typické použití přímého přístupu | 302 |
| 17.10 | Zbývající užitečné funkce ze <code><stdio.h></code> | 306 |
| 17.10.1 | Funkce <code>freopen()</code> – přesměrování proudu | 306 |
| 17.10.2 | Funkce <code>rename()</code> – přejmenování souboru | 308 |
| 17.10.3 | Funkce <code>remove()</code> – zrušení souboru | 308 |
| 17.10.4 | Funkce <code>tmpfile()</code> – pomocný soubor | 308 |
| 17.10.5 | Funkce <code>tmpnam()</code> – jméno pomocného souboru | 308 |
| 17.11 | Ukázka použití funkcí z UNIXové knihovny | 309 |
| 17.12 | Zjištění informací o položkách v adresáři | 311 |
| 17.12.1 | Prostředí UNIXu | 311 |
| | Zjištění pouze jmen položek | 312 |
| | Zjištění typu a stavu položek | 313 |
| 17.12.2 | Prostředí Windows | 316 |
| 18 | Návaznost překladače ANSI C na okolní prostředí | 319 |
| 18.1 | Standardní hlavičkové soubory | 319 |
| 18.2 | Rezervovaná jména | 321 |
| 18.3 | Standardní hlavičkové soubory bez funkčních prototypů | 322 |
| 18.3.1 | Soubor <code><errno.h></code> | 322 |
| 18.3.2 | Soubor <code><limits.h></code> | 323 |
| 18.3.3 | Soubor <code><float.h></code> | 323 |
| 18.3.4 | Soubor <code><stddef.h></code> | 324 |
| 18.4 | Předdefinovaná makra preprocesoru | 325 |
| 18.4.1 | Makro <code>__STDC__</code> | 325 |
| 18.4.2 | Makro <code>__FILE__</code> | 325 |
| 18.4.3 | Makro <code>__LINE__</code> | 325 |
| 18.4.4 | Makro <code>__DATE__</code> | 325 |
| 18.4.5 | Makro <code>__TIME__</code> | 326 |
| 18.5 | Externí proměnné | 326 |
| 18.5.1 | Proměnná <code>errno</code> z <code><errno.h></code> | 326 |
| 18.5.2 | Proměnná <code>sys_errlist</code> ze <code><stdlib.h></code> | 327 |
| 18.5.3 | Proměnná <code>sys_nerr</code> ze <code><stdlib.h></code> | 327 |
| 18.5.4 | Proměnná <code>environ</code> ze <code><stdlib.h></code> | 328 |
| 18.5.5 | Proměnná <code>_iob</code> ze <code><stdio.h></code> | 328 |
| 19 | Popis funkcí a maker ze standardních knihoven | 329 |
| 19.1 | <code><assert.h></code> – makro používané při ladění | 329 |
| 19.2 | <code><ctype.h></code> – práce se znaky | 329 |

| | | |
|---------|--|-----|
| 19.3 | <math.h> – matematické funkce v přesnosti double | 330 |
| 19.4 | <locale.h> – přizpůsobení C národnímu prostředí | 331 |
| 19.5 | <setjmp.h> – umožnění běžně nedovolených skoků | 332 |
| 19.6 | <signal.h> – zpracování signálů | 334 |
| 19.7 | <stdarg.h> – práce s proměnným počtem parametrů | 337 |
| 19.8 | <stdio.h> – funkce pro vstup a výstup | 337 |
| 19.9 | <stdlib.h> – obecně užitečné funkce | 339 |
| 19.9.1 | Konverze řetězců na čísla | 339 |
| | Funkce pro jednoduchý převod desítkových čísel | 339 |
| | Funkce pro náročný převod čísel | 339 |
| 19.9.2 | Generátor pseudonáhodných čísel | 340 |
| 19.9.3 | Funkce pracující s dynamickou pamětí | 341 |
| 19.9.4 | Funkce pro spolupráci s operačním systémem | 341 |
| | Funkce <code>abort()</code> | 341 |
| | Funkce <code>atexit()</code> | 341 |
| | Funkce <code>exit()</code> | 342 |
| | Funkce <code>system()</code> | 343 |
| 19.9.5 | Funkce pro hledání a řazení | 343 |
| | Řazení algoritmem <i>quick-sort</i> | 343 |
| | Binární vyhledávání v seřazeném poli | 344 |
| 19.10 | <string.h> – zpracování řetězců | 348 |
| 19.10.1 | Funkce <code>strspn()</code> | 348 |
| 19.10.2 | Funkce <code>strcspn()</code> | 349 |
| 19.10.3 | Funkce <code>strpbrk()</code> | 349 |
| 19.10.4 | Funkce <code>strtok()</code> | 349 |
| 19.10.5 | Funkce <code>memchr()</code> | 350 |
| 19.10.6 | Funkce <code>memcmp()</code> | 350 |
| 19.10.7 | Funkce <code>memcpy()</code> | 351 |
| 19.10.8 | Funkce <code>memmove()</code> | 351 |
| 19.10.9 | Funkce <code>memset()</code> | 351 |
| 19.11 | <time.h> – práce s datem a časem | 353 |
| 19.11.1 | Funkce <code>clock()</code> | 353 |
| 19.11.2 | Měření doby běhu programu | 354 |
| 19.11.3 | Funkce <code>time()</code> | 356 |
| 19.11.4 | Funkce <code>difftime()</code> | 356 |
| 19.11.5 | Funkce <code>localtime()</code> | 356 |
| 19.11.6 | Funkce <code>asctime()</code> | 357 |
| 19.11.7 | Funkce <code>ctime()</code> | 357 |
| 19.11.8 | Funkce <code>strftime()</code> | 358 |
| 19.11.9 | Funkce <code>mktime()</code> | 359 |

| | | |
|-----------|--|------------|
| 20 | Funkce s proměnným počtem parametrů | 362 |
| 20.1 | Úvodní informace | 362 |
| 20.2 | Základní princip | 363 |
| 20.3 | Standardní hlavičkový soubor <code><stdarg.h></code> | 365 |
| 20.4 | Funkce <code>vprintf()</code> , <code>vfprintf()</code> , <code>vsprintf()</code> | 366 |
| 20.5 | Praktické příklady použití FPPP | 367 |
| 20.5.1 | Typické použití <code>vprintf()</code> , <code>vfprintf()</code> , <code>vsprintf()</code> | 367 |
| 20.5.2 | Přesměrování výstupu za běhu programu | 368 |
| 20.5.3 | Seznam parametrů je ukončený domluvenou hodnotou | 370 |
| 20.5.4 | Počet parametrů je předán v pevném parametru | 372 |
| 20.5.5 | Odstrašující případ zcela netypického použití | 373 |
| 21 | Ladění v jazyce C | 376 |
| 21.1 | Ladicí výpisy na <code>stderr</code> | 378 |
| 21.2 | Využití podmíněného překladač pomocí příkazů C-preprocesoru | 378 |
| 21.2.1 | Využití existence symbolické konstanty | 379 |
| 21.2.2 | Využití hodnoty symbolické konstanty | 380 |
| 21.3 | Self-testy modulů nebo funkcí | 383 |
| 21.4 | Využití makra ze souboru <code><assert.h></code> | 385 |
| 21.4.1 | Praktické použití | 386 |
| 21.4.2 | Zrušení <code>aserc</code> | 389 |
| 21.4.3 | Princip <code>aserc</code> | 389 |
| 21.4.4 | Chybné použití <code>aserc</code> | 390 |
| 21.5 | Prostředky pro ladění dynamických procesů | 392 |
| 22 | Minimální znalosti při programování pod UNIXem | 396 |
| 22.1 | Editor | 396 |
| 22.2 | Překladač a sestavovací program | 397 |
| 22.3 | Kontrola zdrojových textů v C | 399 |
| 22.4 | Odděleně překládané moduly | 401 |
| 22.5 | Ladicí prostředky | 404 |
| 23 | Změny v C podle nového standardu ISO | 406 |
| 23.1 | Stručný přehled norem a nejdůležitějších změn | 406 |
| 23.2 | Milé drobnosti | 407 |
| 23.2.1 | Komentáře do konce řádky | 407 |
| 23.2.2 | Definice kdekoliv v kódu | 408 |
| 23.2.3 | Co bylo zrušeno | 408 |

| | | |
|-----------|--|------------|
| 23.3 | Celočíselné datové typy | 409 |
| 23.3.1 | Logické hodnoty | 409 |
| 23.3.2 | Typ <code>long long int</code> | 410 |
| 23.3.3 | Celočíselné typy s přesně danou šířkou | 410 |
| 23.3.4 | Nové formátové specifikace | 412 |
| 23.4 | Okrajové hodnoty reálných čísel | 413 |
| 23.4.1 | Nekonečno | 413 |
| 23.4.2 | <code>NaN</code> | 413 |
| 23.4.3 | Denormální čísla | 413 |
| 23.4.4 | Podpora okrajových hodnot reálných čísel | 413 |
| 23.5 | Pole | 414 |
| 23.5.1 | Inicializace polí | 414 |
| 23.5.2 | Proměnná délka pole | 415 |
| 23.5.3 | Minimální velikost pole jako parametr funkce | 416 |
| 23.5.4 | Složené literály | 416 |
| 23.6 | Funkce | 416 |
| 23.6.1 | Identifikátor funkce | 417 |
| 23.6.2 | Funkce mohou být vnořené | 418 |
| 23.6.3 | Modifikátor <code>inline</code> | 419 |
| 23.7 | Vícebajtové znaky | 420 |
| 23.7.1 | Kódování a definice proměnných | 421 |
| 23.7.2 | Práce s jednotlivými znaky | 421 |
| 23.7.3 | Práce s řetězci | 422 |
| 23.7.4 | Vstupy a výstupy | 423 |
| 24 | Desatero příkázání pro programátory v C | 425 |
| | Literatura | 430 |
| | Rejstřík | 431 |