Tahák jazyka C

# Operace se stringy <string.h>

**Z´ıska´va´n´ı na´poveˇdy**

## Manua´love´ stra´nky (Unix/Linux)

|  |  |
| --- | --- |
| Na´poveˇda k prohl´ızˇecˇi | **man man** |
| Ovla´da´n´ı: |  |
| dolu˚ o rˇa´dek  – o stra´nku | ↓**, Enter**  **mezera, PgDown** |
| nahoru o rˇa´dek |
| ↑**PgUp** |
| – o stra´nku |  |
| na zacˇa´tek, konec | **Home, End** |
| hleda´n´ı | **/ text** |
| ukoncˇen´ı | **q, Q** |
| Vy´beˇr specificke´ u´rovneˇ |  |

**s**, **cs**, **ct** jsou stringy

de´lka stringu strlen(s)

kop´ıruje **cs** do **s** strcpy(s, cs)

* max. prvn´ıch **n** znaku˚ strncpy(s, cs, n)

prˇipoj´ı **cs** za **s** strcat(s, cs)

* prvn´ıch **n** znaku˚ strncat(s, cs, n)

porovna´ **cs** a **ct** strcmp(cs, ct)

* max. prvn´ıch **n** znaku˚ strncmp(cs, ct, n)

najde prvn´ı vy´skyt znaku **c** v **cs** strchr(cs, c)

* posledn´ı vy´skyt znaku **c** v **cs** strrchr(cs, c)

**b**, **cb**, **ct** jsou bloky pameˇti

kopie **n** bytu˚ z **cb** do **b** memcpy(b, cb, n)

* **b** a **cb** se mohou prˇekry´vat memmove(b, cb, n)

porovna´ **n** bytu˚ z **cb** s **ct** memcmp(cb, ct, n)

man n stranka man 3 printf

najde prvn´ı znak **c** v prvn´ıch **n**

bytech **cb**

memchr(cb, c, n)

Ovla´da´n´ı za´vis´ı na hodnoteˇ promeˇnne´ prostrˇed´ı $MANPAGER,

vypln´ı prvn´ıch **n** bytu˚ **b** hodno- memset(b, c, n)

implicitneˇ je to program less.

## Info stra´nky

Na´poveˇda k prohl´ızˇecˇi man info

tou **c**

# Vstup/Vy´stup <stdio.h>

## Standardn´ı vstup/vy´stup

standardní vstup stdin

standardn´ı vy´stup stdout

standardn´ı chybovy´ vy´stup stderr

|  |  |
| --- | --- |
| konec souboru (konstanta) | EOF |
| nacˇti znak (int) ze stdin  vytiskni znak na stdout | getchar()  putchar(chr) |
| vytiskni forma´tovana´ data  do stdout  – do stringu, pole znaku˚ **s** | printf(”format”, arg1, ...)  sprintf(s, ”format”, |
|  | arg1, ...) |
| cˇti forma´tovana´ data ze stdin | scanf(”format”, |
|  | &prom1, ...) |
| – ze stringu **s** | sscanf(s, ”format”, |
|  | &prom1, ...) |

## Uzˇitecˇne´ programy

|  |  |
| --- | --- |
|  | info info |
| Ovla´da´n´ı: |  |
| jemny´ pohyb  pohyb po stra´nka´ch, kapitola´ch | **Enter,** ←**,** →**,** ↑**,** ↓  **Mezera, Backspace, Del,** |
|  | **PgUp, PgDown** |
| na zacˇa´tek, konec | **Home, End** |
| na´vrat do nevysˇsˇ´ıho menu | **t** |
| do vysˇsˇ´ı u´rovneˇ menu | **u** |
| prˇedchoz´ı polozˇka menu | **p** |
| na´sleduj´ıc´ı polozˇka menu | **n** |
| hleda´n´ı | **/ text** |
| ukoncˇen´ı | **q, Q** |

whatis slovo Prohleda´ databa´zi manua´lovy´ch stra´-

nek a vyp´ısˇe kra´tkou informaci o za- dane´ stra´nce. Vyhleda´va´ stra´nky stej- ne´ho jme´na jako zadane´ slovo.

apropos slovo Jako whatis, ale vyhleda´ vsˇechny

stra´nky, ktere´ ve sve´m jme´neˇ obsahuj´ı zadane´ slovo.

**Souborovy´ vstup/vy´stup**

ukazatel na promeˇn- nou znacˇ´ıc´ı soubor

FILE \*f;

otevrˇen´ı souboru

f = fopen(”jme´no”, ”mo´d”);

mo´dy: **r** (read), **w** (write), **a** (append) zavrˇi soubor fclose(f)

|  |  |
| --- | --- |
| prˇi chybeˇ vrac´ı != 0 | ferror(f) |
| nacˇti znak  zapisˇ znak **c** | fgetc(f)  fputc(c, f) |
| forma´tovany´ za´pis | fprintf(f, ”format”, arg1, |
|  | ...) |
| cˇten´ı ze souboru | fscanf(f, ”format”, &arg1, |
|  | ...) |

cˇti rˇa´dek do **s** (*<* **max** znaku˚) fgets(s, max, f)

## fputs(s, f)

|  |  |  |
| --- | --- | --- |
| **forma´t** | **argument** | **co se vypíše** |
| %d %i  %o  %u  %x %X | int  unsigned int unsigned int unsigned int | zname´nkove´ des´ıtkove´ cˇ´ıslo  osmicˇkove´ cˇ´ıslo des´ıtkove´ číslo hexa číslo |
| %ld  %lo %lu  %lx | long  unsigned long | dlouhe´ desítkove´ číslo  dlouhe´ číslo |

## Alokace pameˇti

|  |  |  |
| --- | --- | --- |
| 15d | int | zarovná´ desítkové´ cˇ´ıslo na  15 m´ıst mezerami, pokud ma´ cˇ´ıslo v´ıce cˇ´ıslic, zarovna´n´ı se ignoruje |
| %e %E  %f %F  %g %G | double  double double | exponencia´ln´ı forma´t rea´l- ne´ho cˇ´ısla  decima´ln´ı forma´t rea´lne´ho cˇ´ısla  jako %e nebo %f podle toho,  co le´pe vypln´ı prostor |
| %10.5f | double | zarovna´n´ı na 10 m´ıst, cˇ´ıslo se  zaokrouhl´ı na 5 desetinny´ch m´ıst  15 m´ıst celkem, 2 desetinna´ m´ısta, forma´tem %f nebo %e podle toho, co se sem le´pe vejde  4 desetinna´ m´ısta, nezarov- na´no |
| %15.2g | double |
| %.4f | double |
| %s | char \* | textovy´ rˇeteˇzec |
| %c | char | znak |
| %% | – | vyp´ısˇe znak % |
| \n \t | – | novy´ rˇa´dek, tabula´tor |

alokace if ((ptr = malloc(size)) == NULL)

{osˇetrˇenı´ chyby}

uvolneˇn´ı free(ptr); //free(NULL) nenı´

pameˇti chyba!

Funkce malloc vrac´ı hodnotu NULL v prˇ´ıpadeˇ, zˇe prˇi alokaci

dosˇlo k neˇjake´ chybeˇ. Nejcˇasteˇjsˇ´ı chybou je nedostatek volne´

pameˇti. Slusˇna´ aplikace prova´d´ı testova´n´ı chyby prˇi KAZˇ DE´ M vola´n´ı funkce malloc!

# Matematicke´ funkce <math.h>

Prˇi pouzˇit´ı

matematicky´ch funkc´ı

je nutne´ (v prostředí

PO-

SIX/UNIX) aplikaci sestavit s matematickou knihovnu. Prˇi prˇe- kladu je nutne´ spousˇteˇt prˇekladacˇ s parametrem -lm (male´ **l** jako

„link“, male´ **m** jako „math library“).

Vsˇechny funkce z matematicke´ knihovny pracuj´ı s rea´lny´mi datovy´mi typy (double).

|  |  |
| --- | --- |
| trigonom. fce.  inverzn´ı trig. fce. arctan(y/x) | sin(x), cos(x), tan(x)  asin(x), acos(x), atan(x) atan2(y, x) |
| expon. a logaritmicke´  fce. | exp(x), log(x), log10(x) |
| mocnina  odmocnina | pow(x, y)  sqrt(x) |
| zaokrouhlení  absolutní hodnota | round(x), ceil(x), floor(x)  fabs(x) |

Prˇ´ıklad:

double a = 657.73578, b = 55.4152565;

printf(”%8.2f + %8.2f = %8.2e\n”,a,b,a+b);

Vy´sledek:

657.74 + 55.42 = 7.13e+02

## Forma´tovany´ vstup (scanf)

|  |  |  |
| --- | --- | --- |
| **forma´t** | **argument** | **co se nacˇte** |
| %d %ld  %o $lo  %x %lx  %h | int\*, long\* int\*, long\* int\*, long\*  short\* | des´ıtkove´ cˇ´ıslo osmicˇkove´ cˇ´ıslo sˇestna´ctkove´ cˇ´ıslo  short |
| %f %lf | float\*,  double\* | rea´lne´ cˇ´ıslo |
| %c | char\* | jediny´ znak |
| %s | char\*, char[] | textovy´ rˇeteˇzec (slovo) |
| %\*d | – | prˇeskocˇ´ı des´ıtkove´ cˇ´ıslo |

Prˇ´ıklad:

int i; float x; char name[50]; scanf(”%2d %f %\*d %2s”, &i, &x, name);

vstup: 56789 0123 45a72

vy´sledek: i = 56; x = 789.0; name = ¨45¨;

# Standardní funkce <stdlib.h>

|  |  |
| --- | --- |
| absolutn´ı hodnota **int n**  absolutn´ı hodnota **long n** | abs(n)  labs(n) |
| pseudona´hodne´ cˇ´ıslo *<*0, RAND MAX)  nastav na´hodne´ sem´ınko | rand()  srand(n) |
| konec programu  spust´ı syste´movy´ prˇ´ıkaz |  |

**Doporucˇene´ nastaven´ı prˇekladacˇe GCC**

Pro zacˇa´tecˇn´ıky je du˚lezˇite´, aby prˇekladacˇ odchyta´val co nejveˇtsˇ´ı mnozˇstv´ı chyb, a aby bylo mozˇne´ prˇelozˇeny´ program krokovat pomoc´ı debuggeru.

prˇeklad gcc -std=c99 -Wall -pedantic -W

-g -o projekt projekt.c

debugger ddd projekt

## Konverze

exit(ko´d); system(”dir”);

|  |  |
| --- | --- |
| string **s** na double  string **s** na int string **s** na long | atof(s) atoi(s)  atol(s) |
| string **s** na double  string **s** na long  string **s** na unsigned long | strtod(s, &endp) strtol(s, &endp, b)  strtoul(s, &endp, b) |