

Textové řetězce v jazyce C

©2023
Petrásek Jan

Obecně

- Na rozdíl od C++, C#, Javy nebo PowerShellu nemá C datový určený pro uložení textového řetězce (string)
- Využívají se jiné datové struktury, ukládající posloupnost znaků
- Nejjednodušším je použití pole datového typu char.
- \0 je tzv. nulový znak, tím musí končit všechny řetězce
- Nulový znak na konci řetězce je vyžadován standardními knihovnami
- Pro práci s řetězcí slouží knihovna string.h

Deklarace v příkladech

- `char text[5] = {'d', 'u', 'h', 'a', '\0'};` //slovo má 4 znaky, 5. znak je ukončovací
- `char text[5] = "duha";` //přirozenější zápis
- `char text[] = "duha";` //délka pole se vypočte automaticky
- `char text[5];` //prázdné pole pro 5 znaků
~~`text = "duha";`~~ //nelze!, syntax error

Výpis

- Využívá se formátovací značka %s
- Př.:
char text[] = "duha";
printf(" Za oknem je %s ", text);

Práce se znaky

Přístup ke znaku

- Pomocí indexu jako u jiného pole
- Př.:

```
char text[] = "den";  
text[0] = 'B';  
printf("%s", text); //vypíše  
Ben
```

Předčasné ukončení

- Proveďte se posunutím nulového znaku
- Př.:

```
char text[] = " ucitelka ";  
text[6] = '\n';  
printf("%s", text); //vypíše  
ucitel
```

Vstup

- Využívá se formátovací značka %s
- Př.:

```
printf("Zadej své jméno: ");  
char jmeno[51];  
scanf("%50s", jmeno);  
printf("Ahoj uživateli %s, vítám tě!", jmeno);
```
- Nenechá napsat jméno s mezerou, přijme vstup pouze po mezeru
- Vždy je nutné do formátovací značky uvést nejprve počet znaků, až pak s

Vstup

- Verze umožňující zápis s mezerou
- Př.:

```
printf("Zadej své jméno: ");  
char jmeno[51];  
scanf("%50 [^\n] s", jmeno);  
printf("Ahoj uživateli %s, vítám tě!", jmeno);
```
- Funkci gets() se vyhněte, jelikož neumožňuje omezit délku načítaného řetězce a fgets() se musí přesměrovat na standardní vstup
- MS Visual studio neumožní provést přesměrování fgets()

Standardní funkce – knihovna string.h

- Zjištění délky řetězce: `strlen()`
- Sloučení dvou řetězců do prvního: `strcat()`
- Zkopírování textového řetězce do nové ho pole: `strcpy()`
- Vyhledání dílčí části řetězce: `strstr()`
- Abecední porovnání dvou řetězců: `strcmp()` (záporné číslo, pokud je druhý před prvním, 0 pokud jsou stejné)
- Jinak lze s textem zacházet jako s polem