

Символьні рядки

Символьний рядок - це масив символів, замкнений у лапки ("). Він має тип `char`. Нульовий символ (`\0`) автоматично додається останнім байтом символного рядка та виконує роль ознаки його кінця. Кількість елементів у масиві дорівнює кількості символів у рядку плюс один, оскільки нульовий символ також є елементом масива.

Оголошення символних рядків

Масиви символів оголошуються так:

```
char назва_рядка[довжина_рядка];
```

Наприклад: `char a[32];`

```
const int len=80; char str[len];
```

Оскільки рядки є масивами символів, то назва рядка є вказівником на його перший елемент.

Символ `NULL` є одним з символів рядка. Тобто `a` – масив з елементами, пронумерованими `a[0]...a[31]`

У програмах на `C++` зустрічаються окремі символи, взяті в одинарні лапки (`'F'`), і символи у подвійних лапках (`"F"`). `'F'` – символна константа, 1 байт пам'яті; `"F"` – рядкова константа, 2 байти пам'яті.

Ініціалізація символьного рядка

C++ дозволяє ініціалізувати масиви при оголошенні:

```
const char text[]="Ми вивчаємо програмування";
```

```
char slovo[11]="University";
```

```
char sl[]={ 'U', 'n', 'i', 'v', 'e', 'r', 's', 'i', 't', 'y', '\0' };
```

```
char tit[32]="Hello, C++";
```

Якщо кількість символів, що присвоюється рядку, менша від розміру масиву, більшість компіляторів C++ присвоюють символи NULL елементам рядкового масиву, що залишаються. Пізніше, в програмі, у цей масив можна буде вмістити довший рядок, обов'язково помістивши в кінці символ '\0'.

Введення та виведення рядків

Для вводу і виводу символьних даних у бібліотеці мови C (файл <stdio.h>) визначені наступні функції:

- `int getchar()` - здійснює введення одного символу з вхідного потоку і повертає один байт інформації (символ) у вигляді значення типу `int`. Це робиться для розпізнавання ситуації, коли при зчитуванні буде досягнуто кінець файлу.
- `int putchar (int c)` – розміщує в стандартний вихідний потік символ `c`.

Приклад:

```
# include <stdio.h>
void main()
{char c, d;
c=getchar(); putchar(c);
d=getchar(); putchar(d);
}
```

- `char* gets(char*s)` – зчитує рядок `s` із стандартного потоку до появи символу `'\n'`, сам символ `'\n'` у рядок не заноситься. Повертає вказівник на цей рядок.
- `int puts(const char* s)` – записує рядок у стандартний потік виводу, додаючи в кінці рядка символ `'\n'`, у випадку вдалого завершення повертає значення більше або рівне 0 і від'ємне значення у випадку помилки.

Також для вводу/виводу рядка можна використовувати функції `scanf/printf`, відповідно, задавши специфікатор формату `%s`:

```
# include <stdio.h>
void main(){
const int n=10;
char s[n];
scanf("%s", s); printf("%s", s);}

```

Бібліотечні функції для роботи з рядками

Прототипи функцій для роботи з символьними рядками знаходяться у файлі `string.h`. Тому для їх використання необхідна директорія препроцесора `#include<string.h>`

1) функції для об'єднання символьних рядків

- `char *strcat(char *s1, char *s2);` - за рядком `s1` розміщує рядок `s2`.

Наприклад, є два рядки символів

```
char *s1= "Ракетний";
```

```
char *s2= " двигун";
```

Результатом виконання функції `strcat(s1, s2)` буде рядок "Ракетний двигун"ів

- `char *strncat(char *s1, char *s2, int n);`

Наприклад, `strncat(s1, s2, 5);`

За рядком `s1` буде розміщено 5 перших символів з рядка `s2`.

2) функції копіювання рядків

- `char *strcpy(char *s1, char *s2);`

Повертає вказівник на перший символ рядка, отриманий в результаті копіювання символів рядка `s2` в `s1` до першого символу `'\0'`.

Наприклад, `strcpy(s1, s2)`. Рядок `s2` копіюється в `s1`.

- `char *strncpy(char *s1, char *s2, int n);`

Перші `n` символів з рядку `s2` копіюються в рядок `s1`.

3) функція визначення довжини рядка

`int strlen(char *s);`

Наприклад, `int m;`

`char *s= "Колобок";`

`m=strlen(s);`

4) функція порівняння символічних рядків

- `int strcmp(char *s1, char *s2);`

Порівнюємо два рядки. Якщо вони співпадають, то функція повертає нуль. Якщо рядок `s1 < s2`, то повертається від'ємне ціле число. У випадку `s1 > s2` повертається число більше нуля.

Порівняння відбувається за кодами в таблиці ASCII.)

- `int strncmp(char *s1, char *s2, int n);`

Порівнюються перші `n` символів рядків `s1` та `s2`.

5) функція визначення позиції першого входження заданого символу в рядок

`char *strchr(char *s, char c);`

Якщо заданий символ відсутній, то повертається нуль.

Приклад використання функції порівняння рядків

```
#include<stdio.h>
#include<string.h>
void main()
{
char *s1= "Stroka1";
char *s2= "Stroka2";
int n;
n=strcmp(s1,s2);
printf("n=%i\n",n); //n=0
s2="Stroka2";
n=strcmp(s1,s2);
printf("n=%i\n", n); //n= -1
s2= "Stroka 0";
n=strcmp(s1.s2);
printf("n=%i\n", n); //n=1
n=strncmp(s1,s2,4); //Сравнение первых 4 символов
printf("n=%i\n",n); //n=0
}
```