

Символни низове

Терминираща нула.

Всеки валиден стрингов литерал има **терминираща нула** накрая

Пример: *"fmi"* е всъщност масив от символи с дължина 4

{'f', 'm', 'i', '\0'}

Създаване на символен низ.

- Константен символен низ, т.е. низ, който няма да бъде променен се създава по следния начин:

```
const char* string = "this is a string";
```

- Статично заделен символен низ, който може да бъде променен впоследствие:

```
const int MAX = 128;  
char str[MAX] = { 'f', 'm', 'i', '\0' }; // OK  
char str[MAX] = "fmi"; // НЕ е OK!
```

```
char str[MAX]; // OK, може да му бъде зададена  
                // стойност чрез функцията strcpy.  
strcpy(str, "fmi"); // Повече за strcpy по-надолу
```

- Динамично заделен символен низ, който може да бъде променен впоследствие:

```
char* str = new (std::nothrow) char[MAX];  
// Отново чрез strcpy можем да зададем стойност
```

Четене от конзолата.

- Чрез cin

```
std::cin >> str;  
// Прочита символи от конзолата до срещане на празен  
// символ (разстояние, табулация, нов ред и други)
```

- Чрез cin.getline()

```
std::cin.getline(str, MAX);  
// Прочита символи от конзолата до срещане на нов ред
```

```
std::cin.getline(str, MAX, 'p');  
// Прочита символи от конзолата до срещане на символ p
```

Забележка: Използването на getline след cin, може да доведе до неприятности.

Извеждане на конзолата.

- Чрез cout
std::cout << str;
// Извежда символите на масива (низа) str до срещане
// на терминираща нула ('\0')

Основни функции за работа със символни низове.

- strcpy(dest, src); // Копира низа src в dest
- strcat(dest, src); // "Долепя" низа src в края на dest
- strlen(str); // Намира и връща дължината на низа str
- strcmp(str1, str2); // Сравнява двата низа str1 и str2

// Забележка: Повече за резултата от strcmp(str1, str2)
// Връща разликата на първият различен символ на низовете.
// Резултатът е **отрицателен**, ако str1 е лексикографски **преди** str2
// Резултатът е 0, ако str1 и str2 имат еднакви символи
// Резултатът е **положителен**, ако str1 е лексикографски **след** str2