# Работа със символни данни. Създаване и инициализиране на символен низ. Операции върху символни низове

В С низовете представляват едномерни масиви от знаци, като в края на масива за приключване на низа се счита слагането на знак „\0“.

# Декларация и инициализация

Когато компилатора „види“ поредица от знаци оградени от двойни кавички автоматично слага знака \0 за да завърши низа.

Примерна декларация:

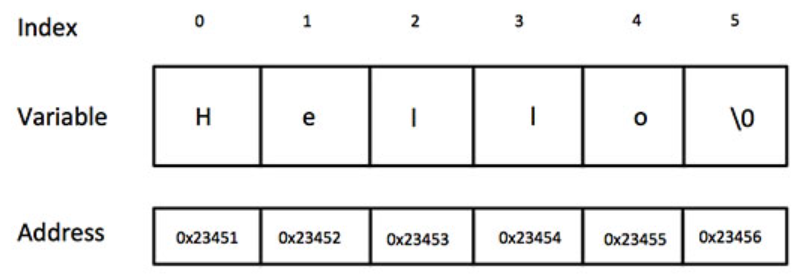
char str [] = “Hello!”;

това е аналогично на:

char str[] = {‘H’,’e’,’l’,’l’,’o’,’\0’};

***Винаги заделяйте място за елемента за край на низ!!!***

Как изглежда това в паметта:



#include <stdio.h>

int main () {

char greeting[6] = {'H', 'e', 'l', 'l', 'o', '\0'};

printf("Greeting message: %s\n", greeting );

return 0;

}

char c[] = "abcd";

char c[50] = "abcd";

char c[] = {'a', 'b', 'c', 'd', '\0'};

char c[5] = {'a', 'b', 'c', 'd', '\0'};

Даване на стойности на низ:

char c[100];

c = "C programming"; // Error! array type is not assignable.

В езика С не може да присвояваме стойност на низ. За целта използваме функции или работим като обикновен масив.

Четене на стойност от клавиатурата:

#include <stdio.h>

int main()

{

char name[20];

printf("Enter name: ");

scanf("%s", name);

printf("Your name is %s.", name);

return 0;

}

Четене на цял ред:

#include <stdio.h>

int main()

{

char name[30];

printf("Enter name: ");

fgets(name, sizeof(name), stdin); // read string

printf("Name: ");

puts(name); // display string

return 0;

}

Подаване на низ на функция:

#include <stdio.h>

void displayString(char str[]);

int main()

{

char str[50];

printf("Enter string: ");

fgets(str, sizeof(str), stdin);

displayString(str); // Passing string to a function.

return 0;

}

void displayString(char str[])

{

printf("String Output: ");

puts(str);

}

Пример с използване на указатели и низове:

#include <stdio.h>

int main(void) {

char name[] = "Harry Potter";

printf("%c", \*name); // Output: H

printf("%c", \*(name+1)); // Output: a

printf("%c", \*(name+7)); // Output: o

char \*namePtr;

namePtr = name;

printf("%c", \*namePtr); // Output: H

printf("%c", \*(namePtr+1)); // Output: a

printf("%c", \*(namePtr+7)); // Output: o

}

Често използвани функции:

**strcpy(s1, s2); -** копира низ 2 в низ 1

**strcat(s1, s2); -** залепя низ2 в края на низ1 като трябва да има място в низ 1 за всички знаци

**strlen(s1); -** връща дължината на низа

**strcmp(s1, s2); -** сравнява двата низа лексикографски. Връща 0 при еднакви, <0 при s1<s2 и >0 при s1>s2.

**strchr(s1, ch); -** връща указател към първото срещане на знака ch в s1

**strstr(s1, s2); -** връща указател към първото срещане на s2 в s1.

Пример за ползване на функциите:

#include <stdio.h>

#include <string.h>

int main () {

char str1[12] = "Hello";

char str2[12] = "World";

char str3[12];

int len ;

/\* copy str1 into str3 \*/

strcpy(str3, str1);

printf("strcpy( str3, str1) : %s\n", str3 );

/\* concatenates str1 and str2 \*/

strcat( str1, str2);

printf("strcat( str1, str2): %s\n", str1 );

/\* total lenghth of str1 after concatenation \*/

len = strlen(str1);

printf("strlen(str1) : %d\n", len );

return 0;

}

Налични са и други функции, които може да разгледате на адрес: <http://cplusplus.com/reference/cstring/>

Задачи:

1. Напишете функция, която намира дължината на низ без да ползвате вградените.
2. Напишете програма, която намира броя на думите в низ
3. Напишете програма, която изкарва колко пъти се среща всяка буква в низ
4. Напишете програма, която сравнява два низа без да ползвате вградените функции
5. Напишете функция, която връща броя на гласните в низ.
6. Напишете програма, която заменя малките букви в низ с големи