

C Strings



- String is a collection of characters in a linear sequence, which is terminated with a null character '\0' at the end by-default.
 - A single character is represented in single quotes as 'a'.
 - A string is represented in double quotes as "hello".

- Declaration: `char s[5];`

s[0]	s[1]	s[2]	s[3]	s[4]

- Initialization:

```
char c[] = "abcd";
char c[50] = "abcd";
char c[] = {'a', 'b', 'c', 'd', '\0'};
char c[5] = {'a', 'b', 'c', 'd', '\0'};
```

c[0]	c[1]	c[2]	c[3]	c[4]
a	b	c	d	\0

- Assignment:

```
char c[100];
c = "C programming"; // Error! array type is not assignable.
```

NOTE – String can't be assigned directly to a variable, use strcpy().

C Strings

Q. How to read a string from the user?

- scanf() function can be used to read a string, but it only reads the sequence of characters until it encounters whitespace (space, newline, tab etc.)

```
#include <stdio.h>
int main()
{
    char name[20];
    printf("Enter name: ");
    scanf("%s", name);
    printf("Your name is %s.", name);
    return 0;
}
```

Output

```
Enter name: Dennis Ritchie
Your name is Dennis.
```

C Strings



Q. How to read a line from a text?

- gets() function is used to read a line of string, and puts() is used to display the string.

```
#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;
}
```

NOTE : gets() is removed from C Standard because it allows you to input any length of characters, which may lead to buffer overflow.

C Strings



- Passing Strings to a function:

```
#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);
}
```

C Strings



- Built-in String Library Functions:
 - C supports large number of string built-in functions which are declared in `<string.h>` header file, and using them we can do the string manipulations.
 - Following are some if the widely used built-in string functions:

Function	Work of Function
<code>strlen()</code>	computes string's length
<code>strcpy()</code>	copies a string to another
<code>strcat()</code>	concatenates(joins) two strings
<code>strcmp()</code>	compares two strings
<code>strlwr()</code>	converts string to lowercase
<code>strupr()</code>	converts string to uppercase