

Character Arrays {Strings}

By

Prof. Muhammad Iqbal Bhat

Department of Higher Education
Government Degree College Beerwah

Topics:

1

What are strings

2

Declaration and
initialization of
strings

3

String functions

What are Strings?



Characters are the fundamental building blocks of your programs.



Every program is composed of characters that—when grouped together meaningfully—the computer interprets as a series of instructions used to accomplish a task.



A string is a series of characters treated as a single unit.



A string may include letters, digits and various special characters such as +, -, *, / and \$.



String literals, or string constants, are written in double quotation marks as follows:

"John Q. Doe"	(a name)
"99999 Main Street"	(a street address)
"Waltham, Massachusetts"	(a city and state)
"(201) 555-1212"	(a telephone number)

Declaration and Initialization of Strings?



In C, we declare a string using the `char` data type and an array of characters. (*e.g.* `char str[20];`)



We can also initialize a string at the time of declaration using a string literal. A string literal is a sequence of characters enclosed in double quotes. `char greeting[] = "Hello, world!";`



We can also initialize individual elements of a string array using an index, like this: `char str[6] = {'h', 'e', 'l', 'l', 'o', '\0'};`



Every string must end with the **null character** (`'\0'`). Printing a "string" that does not contain a terminating null character is a logic error.

Initializing char Arrays and char * Pointers

You can initialize a character array or a char * variable with a string.

```
char color[] = "blue";  
const char *colorPtr = "blue";
```

String Literals Should Not Be Modified

Reading a String with scanf

```
scanf("%19s", word);
```

```
char name[20];  
printf("Enter your name: ");  
scanf("%s", name);  
printf("Hello, %s!", name);
```



String handling functions:

Prof. M. Iqbal Bhat (JKHED)

String handling functions?

C provides a number of string-handling functions that are defined in the `<string.h>` header file.

These functions can be used to manipulate strings in various ways, such as finding the length of a string, copying or concatenating strings, or comparing two strings.

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

1. strlen() function:

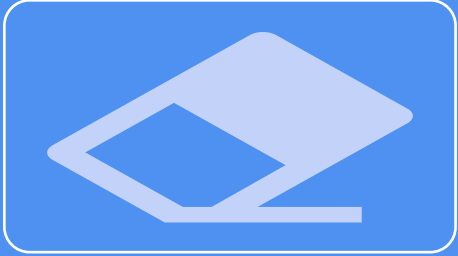


The **strlen()** function is used to find the length of a string, i.e., the number of characters in the string excluding the null terminator character.

```
char str[] = "hello";  
int len = strlen(str); // len is 5
```

Prof. M. Iqbal Bhat (JKUED)

2. strcpy() function:

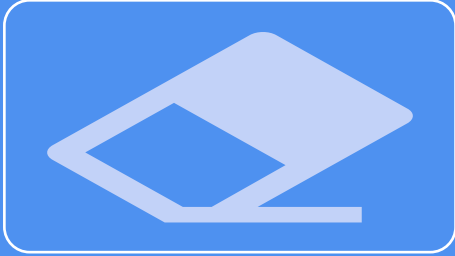


The **strcpy()** function is used to copy one string to another.

```
char src[] = "hello";  
char dest[6];  
// make sure dest has enough space to hold the source string  
strcpy(dest, src); // dest now contains "hello"
```

Prof. M. Iqbal Bhat (JKHED)

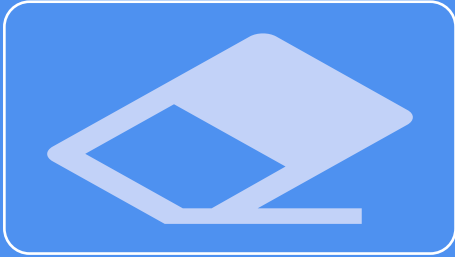
3. strcat() function:



The **strcat()** function is used to concatenate two strings, i.e., to append one string to another.

```
char str1[] = "hello";  
char str2[] = "world";  
char result[12];  
// make sure result has enough space to hold both strings plus the null  
terminator  
strcpy(result, str1);  
strcat(result, str2); // result now contains "helloworld"
```

4. strcmp() function:



The **strcmp()** function is used to compare two strings. It returns an integer value that indicates whether the first string is less than, equal to, or greater than the second string, based on the ASCII values of their characters.

```
#include <stdio.h>
#include <string.h>
int main() {
    char str1[] = "hello";
    char str2[] = "world";
    int result = strcmp(str1, str2);
    if (result < 0) {
        printf("%s is less than %s\n", str1, str2);
    } else if (result > 0) {
        printf("%s is greater than %s\n", str1, str2);
    } else {
        printf("%s is equal to %s\n", str1, str2);
    }
    return 0;
}
```

Manual Program for strlen() function

```
#include <stdio.h>

int main(void) {
    char name[] = "Muhammad Iqbal Bhat";
    printf("Length of String %s is
%d\n", name, find_length(name));
    return 0;
}

int find_length(char *str){
    int length=0;
    while (*str != '\0'){
        length++;
        str++;
    }
    return length;
}
```

Manual Program for strcat() function

```
#include <stdio.h>
int main() {
    char s1[100] = "programming ", s2[] = "is awesome";
    int length, j;

    // store length of s1 in the length variable
    length = 0;
    while (s1[length] != '\0') {
        ++length;
    }

    // concatenate s2 to s1
    for (j = 0; s2[j] != '\0'; ++j, ++length) {
        s1[length] = s2[j];
    }

    // terminating the s1 string
    s1[length] = '\0';

    printf("After concatenation: ");
    puts(s1);

    return 0;
}
```



Prof. M. Iqbal Bhat (JKHED)

Questions?

