SRI AKILANDESWARI WOMEN'S COLLEGE, WANDIWASH

C PROGRAMMING

CLASS : I BCA

Mrs.D.TAMILARASI.

Assistant Professor

Department Of BCA

SWAMY ABEDHANADHA EDUCATIONAL TRUST, WANDIWASH

LECTURE CONTENTS

Section-1:Introduction to String
Section-2:Null Character
Section-3:Declaration of string
Section-4:Initialization of String
Section-5:Conclusion

STRINGS

- A string is nothing but the collection of the individual array elements or characters.
- String is enclosed within Double quotes.
- “programming" is a example of String.
- Each Character Occupy 1 byte of Memory.
- Size of “programming“ = 11 bytes
- String is always Terminated with NULL Character (‘\0′).
- char word[20] = “‘p’ , ‘r’ , ‘o’ , ‘g’ , ‘r’ , ‘a’ , ‘m’ , ‘m’ , ‘I’ , ‘n’ , ‘g’ , ‘\0’”

NULL CHARACTER

- NULL Character is also known as string terminating character.
- It is represented by “\0”.
- NULL Character is having ASCII value 0
- NULL terminates a string, but isn’t part of it
- important for strlen() – length doesn’t include the NULL

DECLARATION OF A STRING

- Since we cannot declare string using String Data Type, instead of

  which we use array of type “char” to create String.

- ***Syntax :***

- char String_Variable_name [SIZE] ;

EXAMPLES

- char city[30];
- char name[20];
- char message[50];

RULES FOR DECLARING A STRING

- String / Character Array Variable name should be legal C Identifier.
- String Variable must have Size specified.
- *char city[];*
- Above Statement will cause compile time error.
- Do not use String as data type because String data type is included in later languages such as C++ / Java. C does not support String data type
- When you are using string for other purpose than accepting and printing data then you must include following header file in your code– #include<string.h>

INITIALIZING STRING (CHARACTER ARRAY)

- Process of Assigning some legal default data to String is Called **Initialization of String**.
- A string can be initialized in different ways. We will explain this with the help of an example.
- Below is an example to declare a string with name as str and initialize it with “GeeksforGeeks”.

1. char str[] = "GeeksforGeeks";
2. char str[50] = "GeeksforGeeks";
3. char str[] =
   {'G','e','e','k','s','f','o','r','G','e','e','k','s','\0'};
4. char str[14] =
   {'G','e','e','k','s','f','o','r','G','e','e','k','s','\0'};

STRING EXAMPLE

```
#include <stdio.h>
int main () {
char greeting[6] = {'H', 'e', 'l', 'l', 'o', '\0'};
printf("Greeting message: %s\n", greeting );
 return 0; }
```

When the above code is compiled and executed, it produces the following result −

```
Greeting message: Hello
```

FUNCTIONS OF STRING.H

Function	Purpose	Example	Output
Strcpy();	Makes a copy of a string	strcpy(s1, “Hi”);	Copies “Hi” to ‘s1’ variable
Strcat();	Appends a string totheend of another string	strcat(“Work”, “Hard”);	Prints “WorkHard”
Strcmp();	Compare two strings alphabetically	strcmp(“hi”,“bye”);	Returns -1.
Strlen();	Returns the number of characters in a string	strlen(“Hi”);	Returns 2.
Strrev();	reverses a given string	Strrev(“Hello”);	olleH
Strlwr();	Converts string to lowercase	Strlwr(“HELLO”);	hello
Strupr();	Converts string to uppercase	Strupr(“hello”);	HELLO

STRING COPY (STRCPY)

- strcpy( ) function copies contents of one string into another string.
- **Syntax** *:* strcpy (destination_string , source_string);
- **Example:**-strcpy (str1, str2) – It copies contents of str2 into str1.

  strcpy (str2, str1) – It copies contents of str1 into str2.

- If destination string length is less than source string, entire source string value won’t be copied into destination string. For example, consider destination string length is 20 and source string length is 30. Then, only 20 characters from source string will be copied into destination string and remaining 10 characters won’t be copied and will be truncated.

STRING CONCAT (STRCAT)

- strncat( ) function in C language concatenates (appends) portion of one string at the end of another string.
- **Syntax** *:* strncat (destination_string , source_string, size);
- **Example:-**strncat (str2, str1, 3); – First 3 characters of str1 is concatenated at the end of str2.

- As you know, each string in C is ended up with null character (‘\0’).
- In strncat( ) operation, null character of destination string is overwritten by source string’s first character and null character is added at the end of new destination string which is created after strncat( ) operation.

STRING COMPARE (STRCMP)

- strcmp( ) function in C compares two given strings and returns zero if they are same.
- If length of string1 < string2, it returns < 0 value that is -1.

- ✓ If length of string1 > string2, it returns > 0 value that is 1
- ✓ If length of string1 = string2 it returns 0.

Syntax : strcmp (str1 , str2);strcmp( ) function is case sensitive. i.e, “A” and “a” are treated as different characters.

STRING LENGTH (STRLEN)

- strlen( ) function in C gives the length of the given string.
- **Syntax** : strlen(str);
- strlen( ) function counts the number of characters in a given string and returns the integer value.
- It stops counting the character when null character is found. Because, null character indicates the end of the string in C.

CONCLUSION

string functions in c provide a convenient and efficient way to manipulate strings. these functions, such as strlen(), strcpy(), strcat(), and strcmp(), allow for common string operations like finding their length, copying them, concatenating them, and comparing them.

THANK YOU