**EXPERIMENT NO. 04**

|  |  |
| --- | --- |
| **DATE OF PERFORMANCE:** | **GRADE:** |
| **DATE OF ASSESSMENT:** | **SIGNATURE OF LECTURER/ TTA:** |
|  |  |

**AIM: Implementation of string.**

**THEORY:**

**STRING:**

**String is nothing but an array of characters (OR char data types). While doing programming in C language, you may have faced challenges wherein you might want to compare two strings, concatenate strings, copy one string to another & perform various string manipulation operations. All of such kind of operations plus many more other functions are available in “string.h” header file. In order to use these string functions you must include string.h file in your C program.**

## STRING DECLARATION:

**METHOD 1:**

**char address[]={'T', 'E', 'X', 'A', 'S', '\0'};**

**METHOD 2:**

**The above string can also be defined as –**

**Char address[6]= "TEXAS";**

**or**

**char address[]="TEXAS";**

**In the above declaration NULL character (\0) will automatically be inserted at the end of the string.**

**WHAT IS NULL CHAR “\0”?  
'\0' represents the end of the string. It is also referred as String terminator & Null Character.**

**STRING FUNCTIONS:**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **STRING FUNCTIONS** | **DESCRIPTION** |
| **1** | **[strcat ( )](http://fresh2refresh.com/c/c-strings/c-strcat-function/" \o "C – strcat() function)** | **Concatenates str2 at the end of str1.** |
| **2** | **[strncat ( )](http://fresh2refresh.com/c/c-strings/c-strncat-function/" \o "C – strncat() function)** | **appends a portion of string to another** |
| **3** | **[strcpy ( )](http://fresh2refresh.com/c/c-strings/c-strcpy-function/" \o "C – strcpy() function)** | **Copies str2 into str1** |
| **4** | **[strncpy ( )](http://fresh2refresh.com/c/c-strings/c-strncpy-function/" \o "C – strncpy() function)** | **copies given number of characters of one string to another** |
| **5** | **[strlen ( )](http://fresh2refresh.com/c/c-strings/c-strlen-function/" \o "C – strlen() function)** | **gives the length of str1.** |
| **6** | **[strcmp ( )](http://fresh2refresh.com/c/c-strings/c-strcmp-function/" \o "C – strcmp() function)** | **Returns 0 if str1 is same as str2. Returns <0 if strl < str2. Returns >0 if str1 > str2.** |
| **7** | **[strcmpi( )](http://fresh2refresh.com/c/c-strings/c-strcmpi-function/" \o "C – strcmpi() function)** | **Same as strcmp() function. But, this function negotiates case.  “A” and “a” are treated as same.** |
| **8** | **[strchr ( )](http://fresh2refresh.com/c/c-strings/c-strchr-function/" \o "C – strchr() function)** | **Returns pointer to first occurrence of char in str1.** |
| **9** | **[strrchr ( )](http://fresh2refresh.com/c/c-strings/c-strrchr/" \o "C – strrchr() function)** | **last occurrence of given character in a string is found** |
| **10** | **[strstr ( )](http://fresh2refresh.com/c/c-strings/c-strstr-function/" \o "C – strstr() function)** | **Returns pointer to first occurrence of str2 in str1.** |
| **11** | **[strdup ( )](http://fresh2refresh.com/c/c-strings/c-strdup-function/" \o "C – strdup() function)** | **duplicates the string** |
| **12** | **[strlwr ( )](http://fresh2refresh.com/c/c-strings/c-strlwr-function/" \o "C – strlwr() function)** | **converts string to lowercase** |
| **13** | **[strupr ( )](http://fresh2refresh.com/c/c-strings/c-strupr-function/" \o "C – strupr() function)** | **converts string to uppercase** |
| **14** | **[strrev ( )](http://fresh2refresh.com/c/c-strings/c-strrev-function/" \o "C – strrev() function)** | **reverses the given string** |
| **15** | **[strset ( )](http://fresh2refresh.com/c/c-strings/c-strset-function/" \o "C – strset() function)** | **sets all character in a string to given character** |
| **16** | **[strnset ( )](http://fresh2refresh.com/c/c-strings/c-strnset-function/" \o "C – strnset() function)** | **It sets the portion of characters in a string to given character** |

**STRLEN VS SIZEOF:**  
**strlen returns you the length of the string stored in array, however sizeof returns the total allocated size assigned to the array**.

**PROGRAM-1: WRITE A C PROGRAM TO READ AND WRITE A STRING USING PRINTF & SCANF.**

**#include <stdio.h>**

**#include<conio.h>**

**#include <string.h>**

**void main()**

**{**

**/\* String Declaration\*/**

**char nickname[20];**

**printf("Enter your Nick name:");**

**/\*I am reading the input string and storing it in nickname\*/**

**scanf("%s", &nickname);**

**/\*Displaying String\*/**

**printf("%s",nickname);**

**getch();**

**}**

**OUTPUT:**

**PROGRAM-2: WRITE A C PROGRAM TO READ AND WRITE A STRING USING GETS & PUTS.**

**#include <stdio.h>**

**#include<conio.h>**

**#include <string.h>**

**Void main()**

**{**

**/\* String Declaration\*/**

**char nickname[20];**

**/\* Console display using puts \*/**

**puts("Enter your Nick name:");**

**/\*Input using gets\*/**

**gets(nickname);**

**puts(nickname);**

**getch();**

**}**

**OUTPUT:**

**PROGRAM-3: WRITE A C PROGRAM TO SHOW USE OF STRLEN( ) FUNCTION.**

**#include <stdio.h>**

**#include<conio.h>**

**#include <string.h>**

**void main()**

**{**

**char str1[20] = "university";**

**printf("Length of string str1: %d", strlen(str1));**

**getch();**

**}**

**OUTPUT:**

**PROGRAM-4: WRITE A C PROGRAM TO SHOW USE OF STRNLEN( ) FUNCTION.**

**#include <stdio.h>**

**#include<conio.h>**

**#include <string.h>**

**Void main()**

**{**

**char str1[20] = "university";**

**printf("Length of string str1 when maxlen is 30: %d", strnlen(str1, 30));**

**printf("Length of string str1 when maxlen is 10: %d", strnlen(str1, 10));**

**getch();**

**}**

**OUTPUT:**

**PROGRAM-5: WRITE A C PROGRAM TO SHOW USE OF STRCMP ( ) FUNCTION.**

**#include <stdio.h>**

**#include<conio.h>**

**#include <string.h>**

**Void main()**

**{**

**char s1[20] = "university";**

**char s2[20] = "university";**

**if (strcmp(s1, s2) ==0)**

**{**

**printf("string 1 and string 2 are equal");**

**}else**

**{**

**printf("string 1 and 2 are different");**

**}**

**getch();**

**}**

**OUTPUT:**

**PROGRAM-6: WRITE A C PROGRAM TO SHOW USE OF STRNCMP ( ) FUNCTION.**

**#include <stdio.h>**

**#include <conio.h>**

**#include <string.h>**

**Void main()**

**{**

**char s1[20] = "university";**

**char s2[20] = "M.S.university";**

**/\* below it is comparing first 8 characters of s1 and s2\*/**

**if (strncmp(s1, s2, 8) ==0)**

**{**

**printf("string 1 and string 2 are equal");**

**}else**

**{**

**printf("string 1 and 2 are different");**

**}**

**getch();**

**}**

**OUTPUT:**

**PROGRAM-7: WRITE A C PROGRAM TO SHOW USE OF STRCAT ( ) FUNCTION.**

**#include <stdio.h>**

**#include <conio.h>**

**#include <string.h>**

**void main()**

**{**

**char s1[10] = "Hello";**

**char s2[10] = "World";**

**strcat(s1,s2);**

**printf("Output string after concatenation: %s", s1);**

**getch();**

**}**

**OUTPUT:**

**PROGRAM-8: WRITE A C PROGRAM TO SHOW USE OF STRNCAT ( ) FUNCTION.**

**#include <stdio.h>**

**#include <conio.h>**

**#include <string.h>**

**void main()**

**{**

**char s1[10] = "Hello";**

**char s2[10] = "World";**

**strncat(s1,s2, 3);**

**printf("Concatenation using strncat: %s", s1);**

**getch();**

**}**

**OUTPUT:**

**PROGRAM-9: WRITE A C PROGRAM TO SHOW USE OF STRCPY ( ) FUNCTION.**

**#include <stdio.h>**

**#include <conio.h>**

**#include <string.h>**

**void main()**

**{**

**char s1[30] = "string 1";**

**char s2[30] = "string 2 : I’m gonna copied into s1";**

**/\* this function has copied s2 into s1\*/**

**strcpy(s1,s2);**

**printf("String s1 is: %s", s1);**

**getch();**

**}**

**OUTPUT:**

**PROGRAM-10: WRITE A C PROGRAM TO SHOW USE OF STRNCPY ( ) FUNCTION.**

**#include <stdio.h>**

**#include <conio.h>**

**#include <string.h>**

**void main()**

**{**

**char first[30] = "string 1";**

**char second[30] = "string 2: I’m using strncpy now";**

**/\* this function has copied first 10 chars of s2 into s1\*/**

**strncpy(s1,s2, 12);**

**printf("String s1 is: %s", s1);**

**getch();**

**}**

**OUTPUT:**

**PROGRAM-11: WRITE A C PROGRAM TO SHOW USE OF STRCMPI ( ) FUNCTION.**

**#include <stdio.h>**

**#include<conio.h>**

**#include <string.h>**

**void main( )**

**{**

**char str1[ ] = "fresh" ;**

**char str2[ ] = "refresh" ;**

**int i, j, k ;**

**i = strcmpi ( str1, "FRESH" ) ;**

**j = strcmpi ( str1, str2 ) ;**

**k = strcmpi ( str1, "f" ) ;**

**printf ( "\n%d %d %d", i, j, k ) ;**

**getch();**

**}**

**OUTPUT:**

**PROGRAM-12: WRITE A C PROGRAM TO SHOW USE OF STRCHR ( ) FUNCTION.**

**#include <stdio.h>**

**#include <conio.h>**

**#include <string.h>**

**void main ()**

**{**

**char string[55] ="This is a string for testing";**

**char \*p;**

**p = strchr (string,'i');**

**printf ("Character i is found at position %d\n",p-string+1);**

**printf ("First occurrence of character \"i\" in \"%s\" is" \ " \"%s\"",string, p);**

**getch();**

**}**

**OUTPUT:**

**PROGRAM-13: WRITE A C PROGRAM TO SHOW USE OF STRRCHR ( ) FUNCTION.**

**#include <stdio.h>**

**#include <conio.h>**

**#include <string.h>**

**void main ()**

**{**

**char string[55] ="This is a string for testing";**

**char \*p;**

**p = strrchr (string,'i');**

**printf ("Character i is found at position %d\n",p-string+1);**

**printf ("Last occurrence of character \"i\" in \"%s\" is" \**

**" \"%s\"",string, p);**

**getch();**

**}**

**OUTPUT:**

**PROGRAM-14: WRITE A C PROGRAM TO SHOW USE OF STRSTR ( ) FUNCTION.**

**#include <stdio.h>**

**#include <conio.h>**

**#include <string.h>**

**void main ()**

**{**

**char string[55] ="This is a test string for testing";**

**char \*p;**

**p = strstr (string,"test");**

**if(p)**

**{**

**printf("string found\n" );**

**printf ("First occurrence of string \"test\" in \"%s\" is"\**

**" \"%s\"",string, p);**

**}**

**else printf("string not found\n" );**

**getch();**

**}**

**OUTPUT:**

**PROGRAM-15: WRITE A C PROGRAM TO SHOW USE OF STRDUP ( ) FUNCTION.**

**#include <stdio.h>**

**#include <conio.h>**

**#include <string.h>**

**void main()**

**{**

**char \*p1 = "Raja";**

**char \*p2;**

**p2 = strdup(p1);**

**printf("Duplicated string is : %s", p2);**

**getch();**

**}**

**OUTPUT:**

**PROGRAM-16: WRITE A C PROGRAM TO SHOW USE OF STRLWR ( ) FUNCTION.**

**#include<stdio.h>**

**#include<conio.h>**

**#include<string.h>**

**void main()**

**{**

**char str[ ] = "MODIFY This String To LOwer";**

**printf("%s\n",strlwr (str));**

**getch();**

**}**

**OUTPUT:**

**PROGRAM-17: WRITE A C PROGRAM TO SHOW USE OF STRUPR ( ) FUNCTION.**

**#include<stdio.h>**

**#include<conio.h>**

**#include<string.h>**

**void main()**

**{**

**char str[ ] = "Modify This String To Upper";**

**printf("%s\n",strupr(str));**

**getch();**

**}**

**OUTPUT:**

**PROGRAM-18: WRITE A C PROGRAM TO SHOW USE OF STRREV ( ) FUNCTION.**

**#include<stdio.h>**

**#include<conio.h>**

**#include<string.h>**

**void main( )**

**{**

**char name[30] = "Hello";**

**printf("String before strrev( ) : %s\n",name);**

**printf("String after strrev( ) : %s",strrev(name));**

**getch();**

**}**

**OUTPUT:**

**PROGRAM-19: WRITE A C PROGRAM TO SHOW USE OF STRSET ( ) FUNCTION.**

**#include<stdio.h>**

**#include<conio.h>**

**#include<string.h>**

**void main()**

**{**

**char str[20] = "Test String";**

**printf("Original string is : %s", str);**

**printf("Test string after strset() : %s",strset(str,'#'));**

**printf("After string set: %s",str);**

**getch();**

**}**

**OUTPUT:**

**PROGRAM-20: WRITE A C PROGRAM TO SHOW USE OF STRNSET ( ) FUNCTION.**

**#include<stdio.h>**

**#include<conio.h>**

**#include<string.h>**

**void main()**

**{**

**char str[20] = "Test String";**

**printf("Original string is : %s", str);**

**printf("Test string after string n set" \**

**" : %s", strnset(str,'#',4));**

**printf("After string n set : %s", str);**

**getch();**

**}**

**OUTPUT:**

**PROGRAM-21: WRITE A C PROGRAM TO PASS A STRING TO FUNCTION.**

**#include <stdio.h>**

**#include <conio.h>**

**#include <string.h>**

**void Display(char ch[]);**

**void main(){**

**char c[50];**

**printf("Enter string: ");**

**gets(c);**

**Display(c); // Passing string c to function.**

**Getch();**

**}**

**void Display(char ch[]){**

**printf("String Output: ");**

**puts(ch);**

**}**

**OUTPUT:**