Unit-3 lesson→3 [Strings]

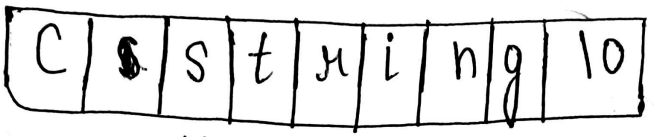
Q-1 What is a Strings?

Ans → In C programming, a string is a sequence of characters terminated with a null character \0.

for example: String is one demensional Character array.

Char c[] = "C String";

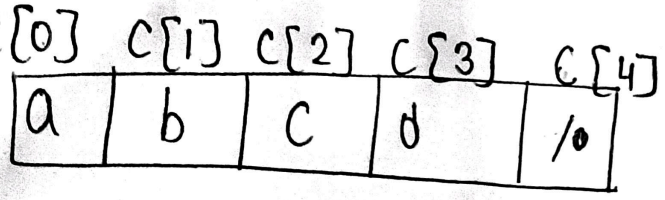
When the compiler encounters a sequence of characters enclosed in the double quotation marks, it appends a null characters \0 at the end by default.



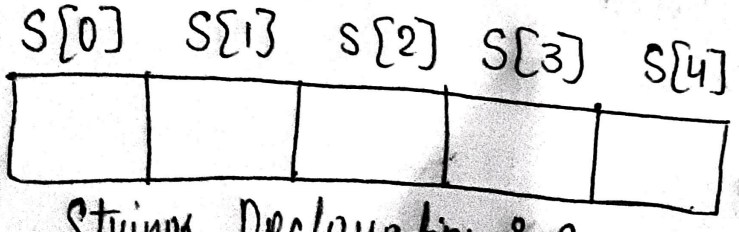
Memory Diagram

- DECLARATION & INITIALIZATION of a STRING?

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




```
Char s[5];
```



Strings Declaration in C.

• Strings

★ Assigning Values to Strings

```
Char c[100];
Strcpy(c, "C programming");
```

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

★ Read String from to User

Example 1: Scanf() to read a String

```
#include <Stdio.h>
int main()
{ Char name[20];
  Printf("Enter name:");
  Scanf("%s", name);
  Printf("Your name is %s", name);
  Return 0;
}
```

★ Read & Display line of Strings

How to read a line of text?

```
#include <Stdio.h>
int main()
{ Char name[30];
  Printf("Enter name:");
  gets(name) // read string printf("Name!");
  PutS(name); // display string
```

return 0;
}

Unit→3 Lesson→4 [Strings functions]

<string.h> is a library File that predefines String related function.

Q- What are String functions?

Ans →

Strcpy()	StrLwr()	→ String.h
Strcat()	Strrev()	
Strlen()	Strcmp()	
Strupr()	~~Strcmpi()~~	

→ reverse strings.

★ Strlen() :- for find the length of string.

Syntax :

Size_t strlen (const char* str)

Size_t represents unsigned short

It returns the length of the string without including end character -

("terminating char. \0")

Example of Strlen :-

```
#include <Stdio.h>
#include <String.h>
int main()
{
  char str1[20] = "Beginner Book";
  Printf (" Length of String str 1 : %d , strlen(str1
  return 0;
}
```

Out put ;

Length of String str 1 : 13

* Strcmp() Compare two strings

It compares the two strings and returns an integer value. If both the strings are same (equal) that this function would return 0 otherwise it may return a negative or positive value based on the comparison.

If String 1 < String 2 OR String 1 is substring of string2 then It would result in a negative value. If String 1 > String 2 then it would return positive value!

If String 1 = = String 2 then would get 0 (zero) when you use this function for Comparie strings.

• Example of Strcmp:

```
#include <Stdio.h>
#include <String.h>
int main()
{ Char S1[20] = "Beginner Book";
  Char S2[20] = "Beginner BookK.Com";
  If (Strcmp (S1, S2) = = 0)
  {
   Print f (" String 1 and String 2 are eg equal");
  } else
    { Print f ("String 1 and 2 are different");
    }
  return 0;
  }
```

Output:
String 1 and 2 are different

* Strcat() Merg two strings.
Char * Strcat(Char * Str1, char * Str2)
It Concatenates two Strings and retuns the Concaten
ated String.

• Example of Strcat:

```
#include<Stdio.h>
#include<String.h>
int main()
{
 Char S1[10] = "Hello";
 Char S2[10] = "World";
 Strcat(S1, S2);
Printf("Output string after Concatenation: %s",S1);
return 0;
}
```

Output:-
Output string after Concatenation: Hello World.

* Strcpy():- Copy one String to another string.

- Char * strcpy (char *Str1, Char* Str2)
It copies the String Str2 into String Str1,
including the end character (terminator char '\0').

• Example of Strcpy:

```
#include <stdio.h>
#include <string.h>
int main()
{
    char s1[30] = "String1";
    char s2[30] = "String2": I'm gonna copied into s1";
    /* this function has copied s2 into s1;
    Strcpy (s1, s2);
    Printf ("String s1, is:%s , s1);
    return 0;
}
```

Output:

String s1 is: String 2: I'm gonna copied in s1

★ Strchr() Character

Char * Strchr (Char * str, int ch)

It searches string (str) for character (ch) while using strchr ch internally gets converted into integer for better searching.

• Example of Strchr:

```
#include<stdio.h>
#include <string.h>
int main()
{ char mystr[30] = "I'm an example of function
                    See strchr";
```

```
    Printf ("%s", strchr (mystr, 'f'));
    return 0;
    }
```

Output :-

f function strchr.

* Strlwr() Convert string upper to lower

The strlwr() function is a built in function in C. and is used to convert a given string into lowercase.

Syntax:

```
Char * strlwr (char *str);
```

Parameter:

- Str: This represents the given string which we want to convert into lowercase.

Example :-

```
#include <stdio.h>
#include <string.h>
int main()
{ Char str[] = "GEEKSFORGEEKS IS THE BEST;
// Converting the given string into lower case.
Printf ("%s\n", strLwr.(str));
return 0;
}
```

☆ Strupr() convert string lower to upper

The Strupr() function is used to converts a given string to uppercase.

Syntax:

Char * Strupr (Char * Str);

Parameter

• Str: This represents the given string which we want to convert into uppercase.

Returns: It returns the modified string obtained after converting the characters of the given string str to Upper Case.

☆Example:

```
#include<stdio.h>
#include<string.h>
int main()
{
Char Str[] = "geeksforgeeks is the best";
// Converting the given into uppercase
Printf ("%s\n", Strupr (Str));
return 0;
}
```

Unit→3 Lesson→5 Programs on arrays & strings

- Pr To find the frequency of characters in a string

```
#include <stdio.h>
int main()
{
 char str[1000], ch;
 int count = 0;
 Printf("Enter a string:");
 fgets(str, sizeof(str), stdin);
 Printf("Enter a character to find its frequency:");
 Scanf("%c", &ch);
 for (int i = 0; str[i] != '\0'; ++i) {
    if (ch == str[i])
        ++count;
  }
  Printf("Frequency of %c = %d", ch, count);
  return 0;
  }
```

- To concatenate two strings

```
#include <stdio.h>
int main()
  {
  char S1[100] = "Programming", S2[] = "is awe
  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;
}
```

- To reverses a string

```
// C program to demonstrate
// Example of strrev() function
#include <stdio.h>
#include <string.h>
int main()
{
Char Str[5];
Printf ("Enter a string");
gets (str);
Printf ("The given string is = %s\n", Str);
Printf ("After reversing string is = %s",
Strrev (Str));
return0;
}
```

- To Copy a string

```
# include< stdio.h>
# in clud < string.h>
int main() {
Char Str1[20] = "C programming" ;
Char Str2[20] ;
// Copying str1 to Str2
  Str cpy (str2, Str1);
Puts (Str2) ; //C programming
 return 0;
}
```

- Program to find the largest number in an array

```
~~# include <stdio.h>~~
int main ()
 {
Int size, i, Largest;
Printf ("\n Enter the size of the array:");
 Scanf ("%d", &size);
int array [size] ; // Declaring array
// Input array elements.
Print f (" \n Enter %d elements of the array :\n", size);
for (i=0; i <size ;i++);
 {
Scan f ("%d", &array [i]);
# include <stdio.h>
}
```

```
// Declaring largest element as the first element
 Largest = array [0];
for (i = 1; i< size; i++)
{
If (largest < array [i])
Largest = array [i];
}
Printf("\n Largest element present in the given
        array is %d", largest);
Return 0;
}
```

- To find Sum of Diagonal Elements of a Matrix.

```
# include <stdio.h>
 int main ()
{ int i, j, rows, columns, a [10] [10], sum = 0;
  Printf ("\n Please Enter number of rows and columns:");
Scanf ("%d %d", &i, &j);
Printf ("\n Please Enter the Matrix elements \n");
  8. for (row = 0; rows< i; rows++)
{ for (columns = 0; columns < j; columns ++)
    {
      Scanf ("%d", &a [row] [columns]);
}
for (row = 0; rows <i; rows++)
{
fo   Sum = Sum + a [rows] [rows];
}
 Printf ("\n In The Sum of Digonal Elements of a matrix
                                    = %d", Sum);
 return 0;
 }
```