**C program 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;

}

Enter name: Dennis Ritchie

Your name is Dennis.

Here, program ignores Ritchie because, scanf() function takes only a single string before the white space, i.e. Dennis.

**C program to read line of text character by character.**

#include <stdio.h>

int main()

{

char name[30], ch;

int i = 0;

printf("Enter name: ");

while(ch != '\n') // terminates if user hit enter

{

ch = getchar();

name[i] = ch;

i++;

}

name[i] = '\0'; // inserting null character at end

printf("Name: %s", name);

return 0;

}

In the program above, using the function getchar(), ch gets a single character from the user each time.

This process is repeated until the user enters return (enter key). Finally, the null character is inserted at the end to make it a string.

This process to take string is tedious.

**Example #3: Using standard library function to read a line of text**

**C program to read line of text using gets() and puts()**

To make life easier, there are predefined functions gets() and puts in C language to read and display string respectively.

#include <stdio.h>

int main()

{

char name[30];

printf("Enter name: ");

gets(name); //Function to read string from user.

printf("Name: ");

puts(name); //Function to display string.

return 0;

}

Both programs have the same output below:

**Output**

Enter name: Tom Hanks

Name: Tom Hanks

## String handling functions

There are various string operations you can perform manually like: finding the length of a string, concatenating (joining) two strings etc.

But, for programmer's ease, many of these library functions are already defined under the header file <string.h>.

Few commonly used string handling functions are discussed below:

| **Function** | **Work of Function** |
| --- | --- |
| [strlen()](https://www.programiz.com/c-programming/library-function/strlen) | Calculates the length of string |
| [strcpy()](https://www.programiz.com/c-programming/library-function/strcpy) | Copies a string to another string |
| [strcat()](https://www.programiz.com/c-programming/library-function/strcat) | Concatenates(joins) two strings |
| [strcmp()](https://www.programiz.com/c-programming/library-function/strcmp) | Compares two string |
| [strlwr()](https://www.programiz.com/c-programming/library-function/strlwr) | Converts string to lowercase |
| [strupr()](https://www.programiz.com/c-programming/library-function/strupr) | Converts string to uppercase |

## Example: Calculate Length of String without Using strlen() Function

#include <stdio.h>

int main()

{

char s[1000], i;

printf("Enter a string: ");

scanf("%s", s);

for(i = 0; s[i] != '\0'; ++i);

printf("Length of string: %d", i);

return 0;

}

**Output**

Enter a string: Programing

Length of string: 10

## Example: Find the Frequency of Characters

#include <stdio.h>

int main()

{

char str[1000], ch;

int i, frequency = 0;

printf("Enter a string: ");

gets(str);

printf("Enter a character to find the frequency: ");

scanf("%c",&ch);

for(i = 0; str[i] != '\0'; ++i)

{

if(ch == str[i])

++frequency;

}

printf("Frequency of %c = %d", ch, frequency);

return 0;

}

**Output**

Enter a string: This website is awesome.

Enter a character to find the frequency: e

Frequency of e = 4

### Example: C strlen() function

#include <stdio.h>

#include <string.h>

int main()

{

char a[20]="Program";

char b[20]={'P','r','o','g','r','a','m','\0'};

char c[20];

printf("Enter string: ");

gets(c);

printf("Length of string a = %d \n",strlen(a));

//calculates the length of string before null charcter.

printf("Length of string b = %d \n",strlen(b));

printf("Length of string c = %d \n",strlen(c));

return 0;

}

**Output**

Enter string: String

Length of string a = 7

Length of string b = 7

Length of string c = 6

## Example

The following example shows the usage of strcpy() function.

#include <stdio.h>

#include <string.h>

int main()

{

char src[40];

char dest[100];

memset(dest, '\0', sizeof(dest));

strcpy(src, "This is tutorialspoint.com");

strcpy(dest, src);

printf("Final copied string : %s\n", dest);

return(0);

}