Course Name: O Level (2nd Sem B2 and B3 Batch) **Subject:** C Language
Topic: String in C Contd.. **Date:** 26-May-2020

- The **gets()** and **puts()** functions facilitate the transfer of strings between the computer and the standard input/output devices.
- Each of these functions accepts a single argument.
- The argument must be a data item that represents a string (e.g. character array).
- The string may include white space characters.
- In the case of gets, the string will enter from the keyboard and will terminate with a new line character (i.e. the string will end when the user presses the RETURN key).

- The **gets()** and **puts()** functions offer simple alternatives to the use of **scanf()** and **printf()** for reading and displaying strings:

Note: gets() & puts( ) handle string, both these functions are defined in "stdio.h" header file.

Example 1: **Program to find the Length of a string without using counter variable.**

```
#include<stdio.h>
#include<conio.h>
void main()
{
int i;
char x[10];
clrscr();
printf("Enter any string\n");
gets(x);
for(i=0;x[i]!='\0';i++);
printf("Length of string is= %d",i);
getch();
}
```

Put the semicolon (;) at the end of the loop, so that the index value of i is increment to '\0' (Null) only, when the loop is matured, then the final value of variable i is length of the string.

Output:

```
Enter your name :Amit Kumar

Length of String is = 10_
```

Example 2: **Write a program to input any string and print it reverse order.**

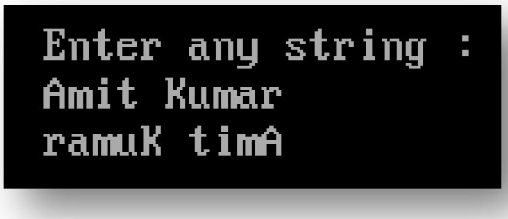
```
#include<stdio.h>
#include<conio.h>
void main()
{
int i;
char x[10];
```

```
clrscr();
printf("Enter any string\n");
gets(x);
for(i=0;x[i]!='\0';i++);
for(i--;i>=0;i--)
printf("%c",x[i]);
getch();
}
```

Put the semicolon (;) at the end of the loop, so that the index value of i is increment to '\0' (Null) only.

**after the above loop is matured,
rewrite the loop and decrease the value of variable i
then again run the loop in decrementing order**

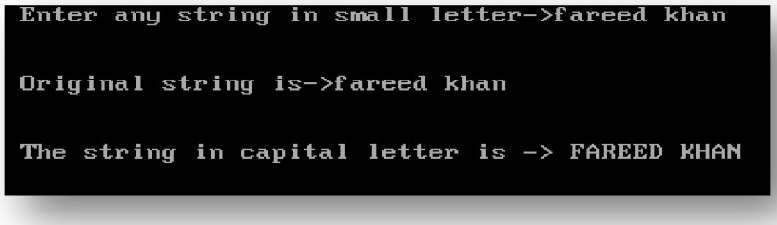
Output:



Example 3: Write a program to input any string in small letter and print it into capital letter.

```
#include<stdio.h>
#include<conio.h>
void main()
{
 int i;
 char str[20];
 clrscr();
 printf("Enter any string in small letter->");
 gets(str);
 printf("Original string is->%s",str);
 for(i=0;i<=strlen(str);i++)
 {
        if(str[i]>=97 && str[i]<=122)
        str[i]=str[i]-32;
 }
 printf("\nThe string in capital letter is -> %s",str);
 getch();
}
```

Output:

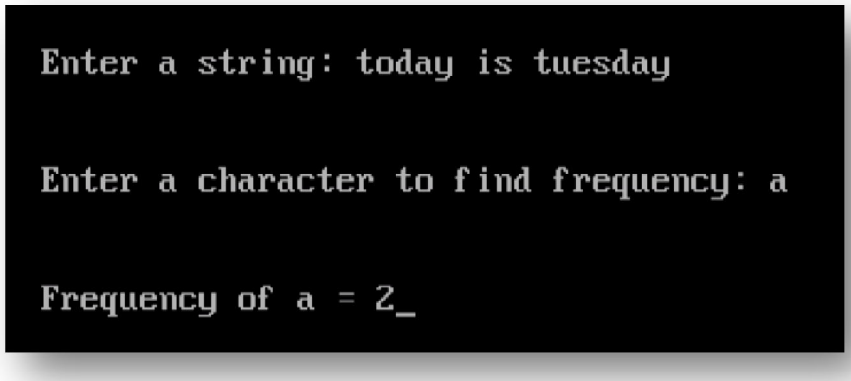


Example 4: Find the frequency of a specified character in given string.

```
#include <stdio.h>
#include <conio.h>
void main()
{
char c[50],ch;
int i,count=0;
clrscr();
printf("\nEnter any string: ");
gets(c);
```

```
printf("\nEnter a character to find frequency: ");
scanf("%c",&ch);
for(i=0;c[i]!='\0';++i)
{
if(ch==c[i])
++count;
}
printf("\nFrequency of %c = %d", ch, count);
getch();
}
```

Output:



Try yourself

1. Write a program to count the number of words in given string.
2. Write a program to copy one string into another string in reverse order then print it.
3. Write a program to input any string in Capital case letter and print it's into small case letter.