Lab no: 7 – STRINGS

Write C programs without using STRING-HANDLING functions for the Questions 1 & 2.

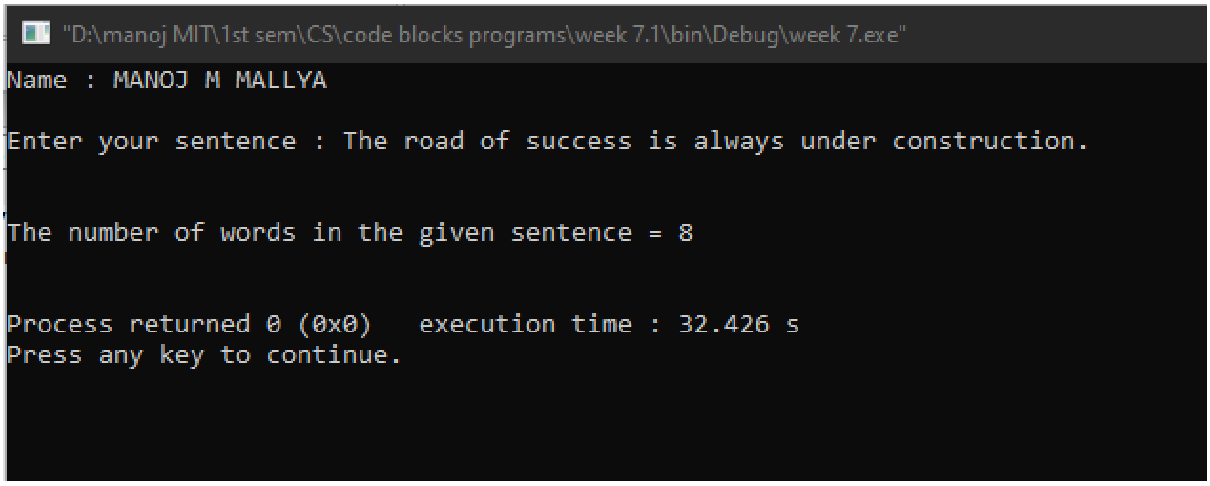
Q1. Count the number of words in a sentence.

Program:

```
//Counting the number of words in a given sentence.
#include <stdio.h>
#include <stdlib.h>

int main()
{
   printf("Name : MANOJ M MALLYA\n\n");
   const int len=100;
   char sent[len];
   int i,count=1;//no.of words in a sentence = no.of spaces encountered + 1
   printf("Enter your sentence : ");
   gets(sent);

   for(i=0;sent[i]!='\0';i++)
   {
      if ((sent[i]==' ') && (sent[i+1]!=' '))
      {
         count++;
      }
   }
```

```
    printf("\n\nThe number of words in the given sentence = %d\n\n",count);
    return 0;
}
```

Output:



"D:\manoj MIT\1st sem\CS\code blocks programs\week 7.1\bin\Debug\week 7.exe"

```
Name : MANOJ M MALLYA

Enter your sentence : The road of success is always under construction.


The number of words in the given sentence = 8


Process returned 0 (0x0)   execution time : 32.426 s
Press any key to continue.
```

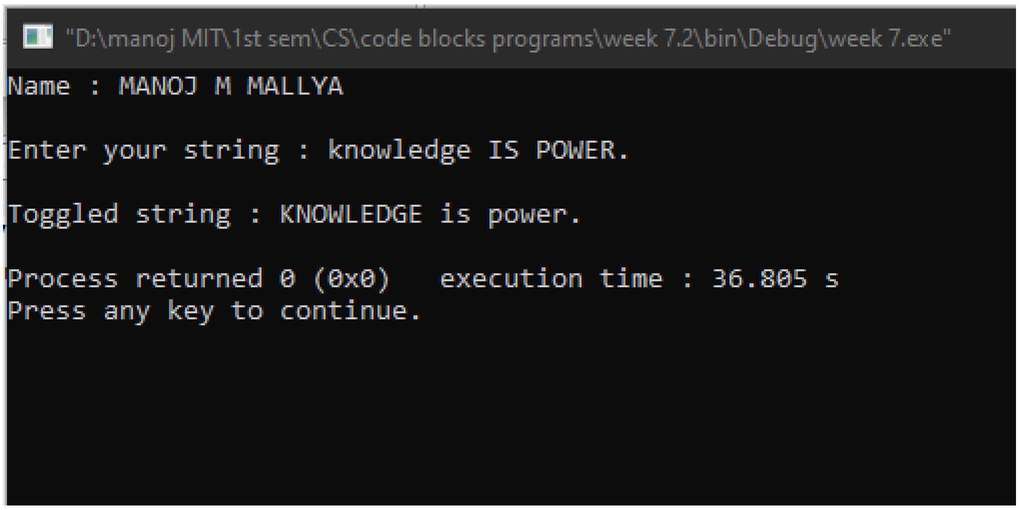
Q2. Input a string and toggle the case of every character in the input string. Ex: INPUT : aBcDe

OUTPUT : AbCdE

Program:

```
//Toggling the case of every character in the input string.
#include <stdio.h>
#include <stdlib.h>

int main()
{
    printf("Name : MANOJ M MALLYA\n\n");
    char str[200];
    int i=0;
    printf("Enter your string : ");
    gets(str);

    while(str[i]!='\0')
    {
        if(str[i]>='A'&&str[i]<='Z')//ASCII value of [A-Z]=[65,90]
        {
            str[i]=str[i]+32;
        }
        else if(str[i]>='a'&&str[i]<='z')//ASCII value of [a-z]=[97,122]
        {
            str[i]=str[i]-32;
        }
        i++;
    }
    printf("\nToggled string : ");
    puts(str);
    return 0;
}
```

Output:



"D:\manoj MIT\1st sem\CS\code blocks programs\week 7.2\bin\Debug\week 7.exe"

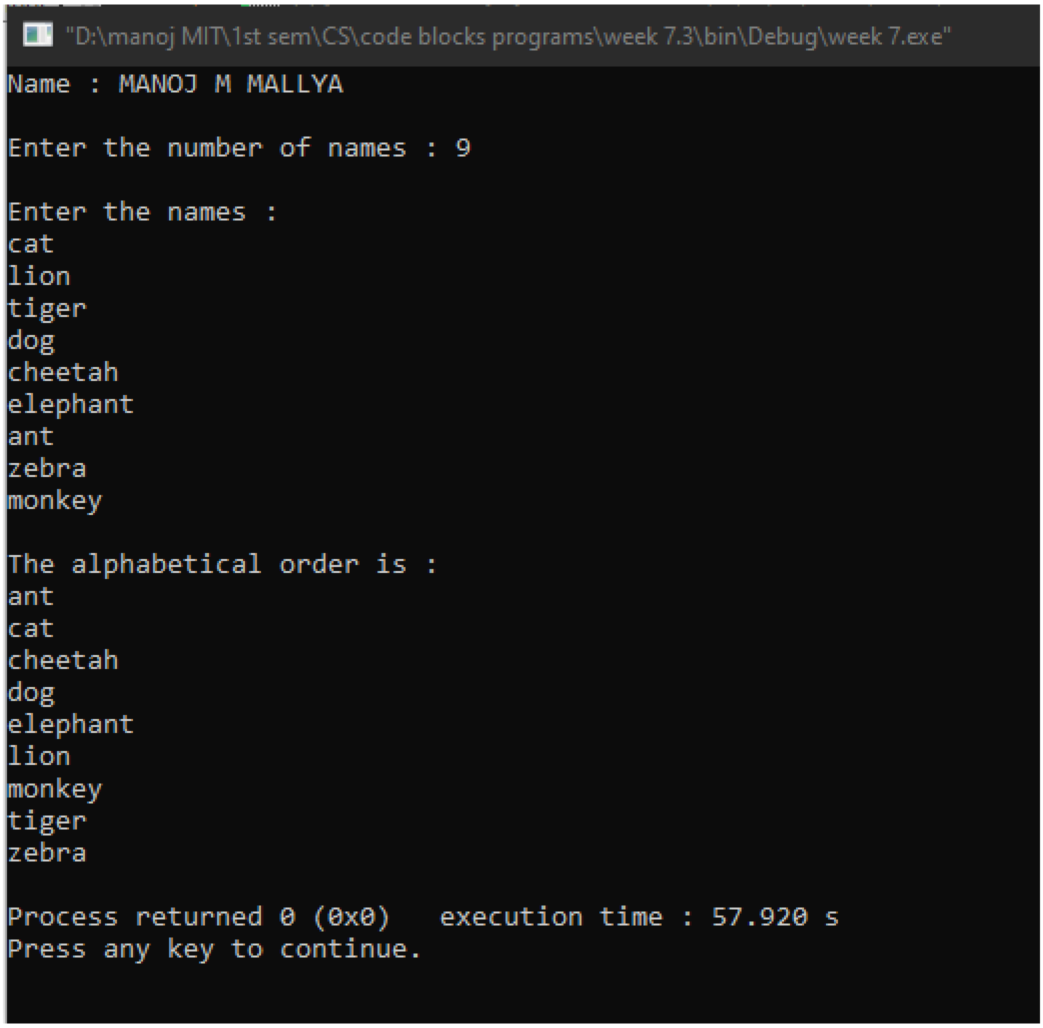
```
Name : MANOJ M MALLYA

Enter your string : knowledge IS POWER.

Toggled string : KNOWLEDGE is power.

Process returned 0 (0x0)   execution time : 36.805 s
Press any key to continue.
```

Q3. Arrange ‘n’ names in alphabetical order (hint: use string handling function-strcpy)

Program:

```
//Arranging ‘n’ names in alphabetical order.


#include <stdio.h>

#include <stdlib.h>

#include <string.h>


int main()

{

    printf("Name : MANOJ M MALLYA\n\n");

    char a[50][20],temp[20];

    int i,j,n;


    printf("Enter the number of names : ");

    scanf("%d",&n);


    printf("\nEnter the names : \n");


    fflush(stdin);
    for(i=0;i<n;i++)//getting the names
    {
        gets(a[i]);
    }

    for(i=0;i<(n-1);i++)//arrangement
    {
        for(j=i+1;j<n;j++)
        {
            if(strcmp (a[i],a[j])>0)//comparing and swapping if necessary
            {
                strcpy(temp,a[i]);
                strcpy(a[i],a[j]);
                strcpy(a[j],temp);
            }
        }
    }

    printf("\nThe alphabetical order is : \n");

    for(i=0;i<n;i++)
    puts(a[i]);

    return 0;
}
```

Output:



"D:\manoj MIT\1st sem\CS\code blocks programs\week 7.3\bin\Debug\week 7.exe"

```
Name : MANOJ M MALLYA

Enter the number of names : 9

Enter the names :
cat
lion
tiger
dog
cheetah
elephant
ant
zebra
monkey

The alphabetical order is :
ant
cat
cheetah
dog
elephant
lion
monkey
tiger
zebra

Process returned 0 (0x0)   execution time : 57.920 s
Press any key to continue.
```

****

Lab no: 8 – MODULAR PROGRAMMING – FUNCTIONS

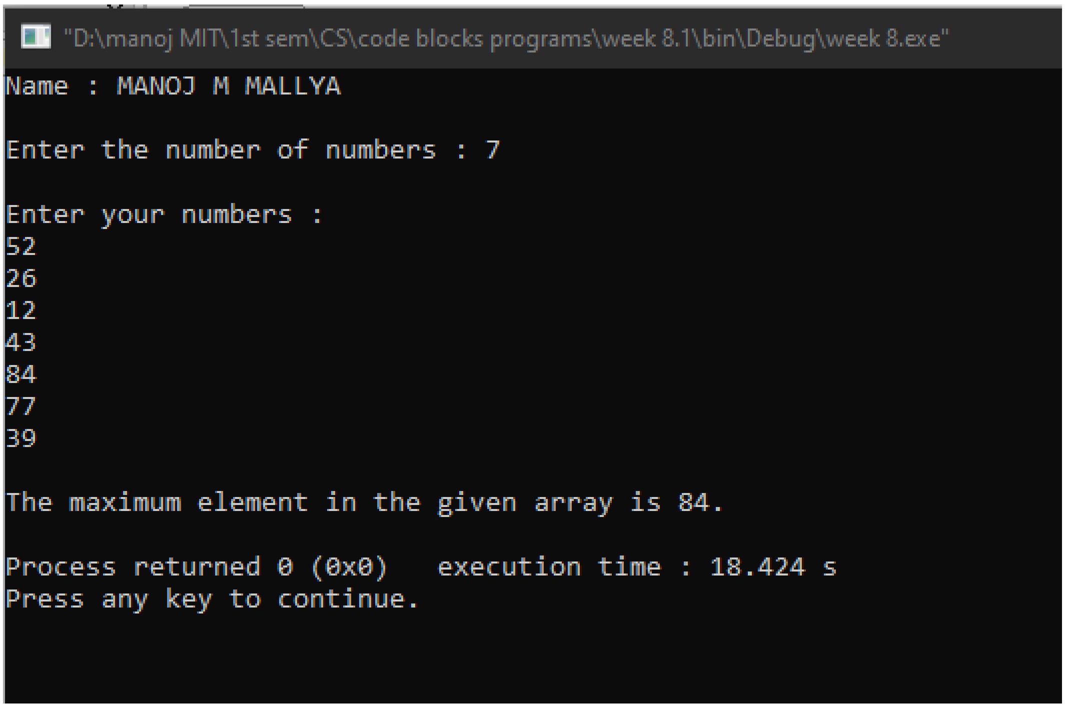
Q4. Write a function Largest to find the maximum of a given list of numbers. Also write a main program to read N numbers and find the largest among them using this function.

Program:

```
//Finding the largest number in a list using functions.
#include <stdio.h>
#include <stdlib.h>

int Largest(int x[],int);//prototype

//function for finding the largest element.
int Largest(int x[100],int s)
{
    int m,i;
    m=x[0];
    for(i=0;i<s;i++)
    {
        if(x[i]>m)
        {
            m=x[i];
        }
    }
    return(m);
    }
```

```
int main()
{
    printf("Name : MANOJ M MALLYA\n\n");
    int a[100],N,i;
    printf("Enter the number of numbers : ");
    scanf("%d",&N);
    printf("\nEnter your numbers : \n");
    for(i=0; i<N; i++)
    {
        scanf("%d",&a[i]);
    }
    printf("\nThe maximum element in the given array is %d.\n",Largest(a,N));
    return 0;
}
```

Output:



"D:\manoj MIT\1st sem\CS\code blocks programs\week 8.1\bin\Debug\week 8.exe"

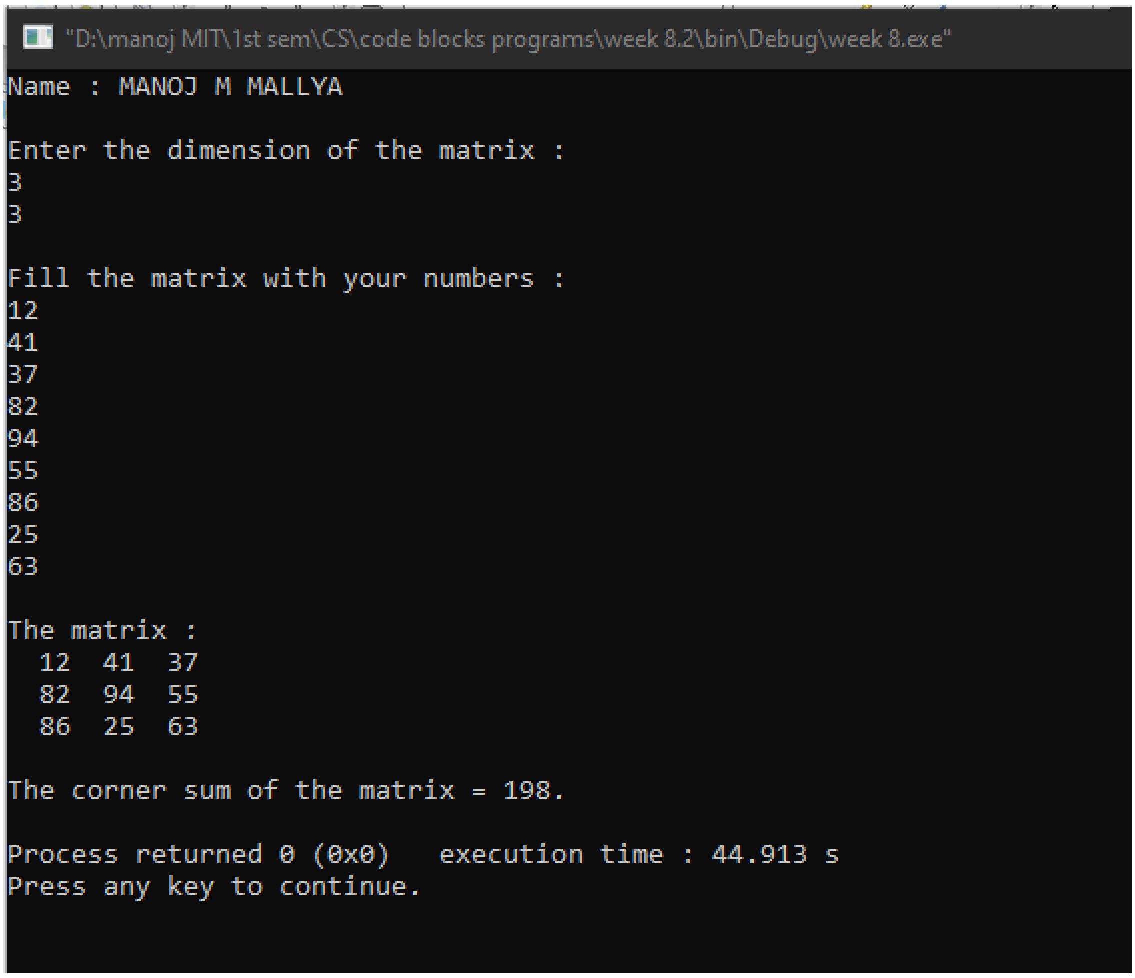
```
Name : MANOJ M MALLYA

Enter the number of numbers : 7

Enter your numbers :
52
26
12
43
84
77
39

The maximum element in the given array is 84.

Process returned 0 (0x0)   execution time : 18.424 s
Press any key to continue.
```

Q5. Write a function CornerSum which takes as a parameter, no. of rows and no. of columns of a matrix and returns the sum of the elements in the four corners of the matrix. Write a main function to test the function.

Program:

```
//Finding the sum of corner elements of a 2D matrix using functions.
#include <stdio.h>
#include <stdlib.h>

int CornerSum(int x[][100],int p,int q);//prototype

int main()
{
    printf("Name : MANOJ M MALLYA\n\n");
    int a[100][100],m,n,i,j;
    printf("Enter the dimension of the matrix : \n");
    scanf("%d %d",&m,&n);//getting the dimensions of the matrix

    printf("\nFill the matrix with your numbers : \n");
    for(i=0;i<m;i++)//getting the matrix
    {
        for(j=0;j<n;j++)
        {
            scanf("%d",&a[i][j]);
        }
    }

    printf("\nThe matrix : \n");
    for(i=0;i<m;i++)//printing the matrix
    {
        for(j=0;j<n;j++)
        {
            printf("%4d",a[i][j]);
        }
        printf("\n");
    }

    printf("\nThe corner sum of the matrix = %d.\n",CornerSum(a,m,n));
    return 0;
}

int CornerSum(int x[][100],int p,int q)
{
    int sum=0;

    //computing the corner element's sum
    for(int i=0;i<p;i++)
    {
        for(int j=0;j<q;j++)
        {
            if((i==0||i==(p-1)) && (j==0||j==(q-1)))
              {
                  sum+=x[i][j];
              }
        }
    }
    return(sum);
}
```

Output:



"D:\manoj MIT\1st sem\CS\code blocks programs\week 8.2\bin\Debug\week 8.exe"

```
Name : MANOJ M MALLYA

Enter the dimension of the matrix :
3
3

Fill the matrix with your numbers :
12
41
37
82
94
55
86
25
63

The matrix :
  12  41  37
  82  94  55
  86  25  63

The corner sum of the matrix = 198.

Process returned 0 (0x0)   execution time : 44.913 s
Press any key to continue.
```

****