Strings:

a. Write a C program to convert a Roman numeral ranging from I to L to its decimal equivalent.

```c
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

void main()
{
  char rom[30];
  int a[30], l, i, k, dec;

  printf("Enter the roman number\n");
  scanf("%s", &rom);

  l =strlen(rom);
  for(i = 0; i < l; i++)
  {
    switch (rom[i])
    {
      case 'I':  a[i] = 1;
              break;
      case 'V':  a[i] = 5;
              break;
      case 'X':  a[i] = 10;
              break;
      case 'L':  a[i] = 50;
              break;
      case 'C':  a[i] = 100;
              break;
      case 'D':  dec = dec + 500;
              break;
      case 'M':  a[i] = 1000;
              break;
      default :  printf("Invalid choice");
              break;
    }
  }
}
```

```c
                        break;
        case 'D':  dec = dec + 500;
                        break;
        case 'M':  a[i] = 1000;
                        break;
        default :  printf("Invalid choice");
                        break;
        }
    }

    k = a[l - 1];
    for(i = l - 1; i > 0; i--)
      {
        if(a[i] > a[i - 1])
          {
            k = k - a[i - 1];
          }
```

```c
        if(a[i] <= a[i - 1])
          {
            k = k + a[i - 1];
          }
      }

    printf("decimal equivalent is %d", k);
}
```

RESULT:

INPUT:
    Enter the roman number
    XIV

OUTPUT:
    decimal equivalent is 14

b. Write a C program that converts a number

ranging from 1 to 50 to Roman equivalent

```c
#include<stdio.h>

void main()
{
   int a,b,e;

   printf("Input a  number (between 1-50):");
   scanf("%d",&e);

  while (e==0||e>50)
    {
       printf ("ERROR: Invalid Input!");
       printf ("\nEnter the number again:");
       scanf ("%d",&e);
    }

a = ((e/10)%10)*10;
b = ((e/1)%10)*1;


if (a == 10)
        printf("X");
else if (a == 20)
        printf("XX");
else if (a == 30)
        printf("XXX");

else if (a == 40)
        printf("XL");
else if (a ==50)
        printf("L");

if (b == 1)
        printf("I");
else if (b == 2)
        printf("II");
else if (b == 3)
        printf("III");
else if (b == 4)
```

```c
                printf("II");
        else if (b == 3)
                printf("III");
        else if (b == 4)
                printf("IV");
        else if (b ==5)
                printf("V");
        else if (b == 6)
                printf("VI");
        else if (b == 7)
                printf("VII");
        else if (b ==8)
                printf("VIII");
        else if (b == 9)
                printf("IX");
}
```

RESULT:

INPUT:
   Input a number (between 1-50): 51

   ERROR: Invalid Input!
   Enter the number again:50

OUTPUT:
   L

c. Write a C program that uses functions to perform the following operations:
   i. To insert a sub-string in to a given main string from a given position.
   ii. To delete n Characters from a given position in a given string.

i. To insert a sub-string in to a given main string from a given position.

```c
#include <stdio.h>
#include <string.h>

void insstr(int);

char MS[20], SS[20], TS[50];
int i,j,pos;

void main()
{
  puts("Enter First String=\n");
  gets(MS);

  puts("Enter Second String=\n");
  gets(SS);

  printf("Enter the position where the item has to be inserted=\n ");
  scanf("%d",&pos);

  insstr(pos);
}
```

```c
void insstr(int pos)
{
  for(i=0;i<pos;i++)
    {
      TS[i]=MS[i];
```

```
void insstr(int pos)
{
  for(i=0;i<pos;i++)
    {
      TS[i]=MS[i];
    }

  for(j=0;SS[j]!='\0';j++)
    {
      TS[i]=SS[j];
      i++;
    }

  for(j=pos;MS[j]!='\0';j++)
    {
      TS[i]=MS[j];
      i++;
    }

  TS[i]='\0';
  printf("The final string is=\n");
  puts(TS);
}
```

RESULT:

INPUT:

Enter First String=
sai

Enter Second String=
ram
Enter the position where the item has to be inserted=
3

OUTPUT:

The final string is=
sairam

ii. To delete n Characters from a given position in a given string.

```c
#include<stdio.h>
#include<string.h>

void delstr(int n,int pos);

char str[15];
int i,j;

void main()
{
   int pos,n;

   printf("Enter the string:\n");
   gets(str);

   printf("Enter the position in string from where to delete");
   scanf("%d", &pos);

   printf("Enter no of char to be deleted");
   scanf("%d",&n);

   delstr(n,pos);
}

void delstr(int n,int pos)
{
  i=pos;

  for(j=pos+n;str[j]!='\0';j++)
   {
     str[i]=str[j];
     i++;
   }
   str[i]='\0';
   printf("\n string after deletion=%s",str);
}
```

```c
        str[i]=str[j];
        i++;
    }
    str[i]='\0';
    printf("\n string after deletion=%s",str);
}
```

RESULT:

INPUT:
    Enter the string:
    kmitcollege
    Enter the position in string from where to delete 4
    Enter no of char to be deleted 7

OUTPUT:
    string after deletion=kmit

d. Write a C program to determine if the given string is a palindrome or not (Spelled
   same in both directions with or without a meaning like madam, civic, noon, abcba,
   etc.)

```c
#include <stdio.h>
#include <string.h>
```

```c
int main()
{
    char str[100];
    int i, len, flag;
    flag = 0;

    printf("\n Please Enter any String :  ");
    gets(str);

    len = strlen(str);

    for(i = 0; i < len; i++)
    {
        if(str[i] != str[len - i - 1])
        {
            flag = 1;
            break;
        }
```

```c
int main()
{
    char str[100];
    int i, len, flag;
    flag = 0;

    printf("\n Please Enter any String :  ");
    gets(str);

    len = strlen(str);

    for(i = 0; i < len; i++)
      {
          if(str[i] != str[len - i - 1])
            {
                flag = 1;
                break;
            }
      }
    if(flag == 0)
      {
          printf("\n %s is a Palindrome String", str);
      }
    else
      {
          printf("\n %s is Not a Palindrome String", str);

      }

    return 0;
}
```

RESULT:

INPUT:
   Please Enter any String : MALAYALAM

OUTPUT:
   MALAYALAM is a Palindrome String

e. Write a C program that displays the position of a character ch in the string S or – 1 if
S doesn't contain ch.

```c
#include<stdio.h>
#include<string.h>
```

```c
void main()
{
   char s[30], t[20];
   char *found;

   puts("Enter the first string: ");
   gets(s);

   puts("Enter the string to be searched: ");
   gets(t);

   found = strstr(s, t);

   if(found)
     {
        printf("Second String is found in the First String
at %d position.\n", found - s);
     }
   else
     {
        printf("-1");
     }
}
```

RESULT - 1:

INPUT:

Enter the first string:
sai ram
Enter the string to be searched:

RESULT - 1:

INPUT:

   Enter the first string:
sai ram
Enter the string to be searched:
ram

OUTPUT:
  Second String is found in the First String at 4
position.

RESULT - 2:

INPUT:
   Enter the first string:
sai ram
Enter the string to be searched:
raj

OUTPUT: -1
f. Write a C program to count the lines, words and characters in a given text.

```c
#include<stdio.h>

void main()
{
   char line[81], a;
   int i,c,

   end = 0,
   characters = 0,
   words = 0,
   lines = 0;

   printf("TYPE ANY  TEXT.\n");
   printf("GIVE ONE SPACE AFTER EACH WORD.\n");
   printf("WHEN COMPLETED, PRESS 'ctrl+z'.\n\n");
```

```c
    lines = 0;

    printf("TYPE ANY  TEXT.\n");
    printf("GIVE ONE SPACE AFTER EACH
WORD.\n");
    printf("WHEN COMPLETED, PRESS
'ctrl+z'.\n\n");

    while( end == 0)
      {
            /* Reading a line of text */
            c = 0;

            while((a=getchar()) != '\n')
                    line[c++] = a;
                line[c] = '\0';

            /* counting the words in a line */
            if(line[0] == '\0')
                        break ;
            else
                {
                    words++;
                    for(i=0; line[i] != '\0';i++)
                            if(line[i] == ' ' || line[i] == '\t')

                                    words++;
                }

            /* counting lines and characters */
            lines = lines +1;
            characters = characters + strlen(line);
      }

    printf ("\n");
    printf("Number of lines = %d\n", lines);
    printf("Number of words = %d\n", words);
    printf("Number of characters = %d\n", characters);
}
```

RESULT:

INPUT:

RESULT:

INPUT:
TYPE ANY TEXT.
GIVE ONE SPACE AFTER EACH WORD.
WHEN COMPLETED, PRESS 'ctrl+z'.

hi how r u
what r u doing
n where r u

^Z

OUTPUT:
Number of lines = 3
Number of words = 12
Number of characters = 35