Lab5

Theory:

The **string.h** header defines one variable type, one macro, and various functions for manipulating arrays of characters. Strings in array are defined as an array of character. The difference between an array and a string is the string is terminated with a special character'\0'.

Some of the string functions that we used in this lab report are described below:

- **strcmp:** strcmp() is a built-in library function and is declared in <string.h> header file. This function takes two string as arguments and compare these two strings lexicographically.
- **strcpy:** strcpy() is a standard library function in C++ and is used to copy one string to another. In C present in string.h header file and in C++ it is present in cstring header file.
- **strlen:** The strlen() function calculates the length of a given string. The strlen() function is defined in string.h header file. It doesn't count null character '\0'

Methodology:

This was the 1st code in which we used **string.h** as a function. The 1st code we did in this lab report was righting our name in upper case and it will convert that name into lower case. The error we faced in this code was we were writing %d in scanf function. So we corrected by replacing %d with %s. The 2nd code we did was to write our names randomly and the output we will get will be in alphabetical order. In this code the was perfectly written but some minor errors that I did in this code was writing stcmp as stemp. I corrected that error by typing the right syntax. The 3rd code we did was wriing the number in decimal form and the output we will get will be in binary form. In this code I did some minor mistakes like typing the rem as tem. This code was also corrected by writing the correct wors. The final code we did was wheather a number was integer or a float number. This code was perfectly written and no any mistakes were done while writing the code.

Objectives:

1. To be familiar with syntax and structure of C-programming.
2. To learn problem solving techniques using C.
3. To learn the basic of string function.

Programs:

- Find out the errors and output of the following programs.

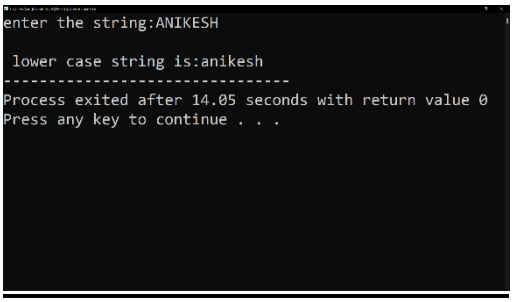
Code:

// Following codes are written and compiled in DevC++

Program 1:

```
#include<stdio.h>
#include<string.h>
int main()
{
    /* this array can hold a string upto 25
    *chars,if you are going to enter large string
    *then increase the array size accordingly
    */
    char str[25];
    int i;
    printf("enter the string:");
    scanf("%s",str);
    for(i=0;i<=strlen(str);i++){
        if(str[i]>=65&&str[i]<=90)
        str[i]=str[i]+32;
    }
    printf("\n lower case string is:%s",str);
    return 0;
}
```

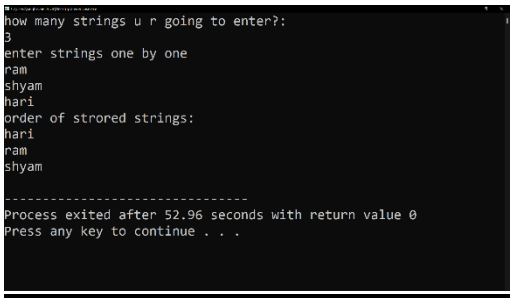
Output:



Program 2:

```
#include<stdio.h>
#include<string.h>
int main()
{
    int i,j,count;
    char str[25][25],temp[25];
    puts("how many strings u r going to enter?:");
    scanf("%d",&count);
    puts("enter strings one by one ");
    for(i=0;i<=count;i++)
    gets(str[i]);
    for(i=0;i<=count;i++)
    for(j=i+1;j<=count;j++){
        if(strcmp(str[i],str[j])>0){
            strcpy(temp,str[i]);
            strcpy(str[i],str[j]);
            strcpy(str[j],temp);
        }
    }
    printf("order of strored strings:");
    for(i=0;i<=count;i++)
    puts(str[i]);
    return 0;}
```

Output :

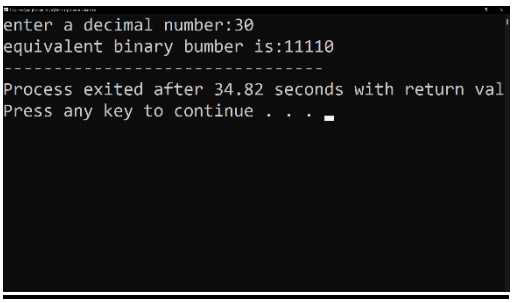


Program 3:

```
#include<stdio.h>
#include<math.h>
long decimalToBinary(int decimalnum)
{
    long binarynum=0;
    int rem,temp=1;
    while(decimalnum!=0)
    {
        rem=decimalnum%2;
        decimalnum=decimalnum/2;

    binarynum=binarynum+rem*temp;
        temp=temp*10;
    }
    return binarynum;
}
    int main(){
        int decimalnum;
        printf("enter a
decimal number:");
        scanf("%d",&decimalnum);
    printf("equivalent    binary bumber
is:%d",decimalToBinary(decimalnum));
        return 0;
}
```

Output :

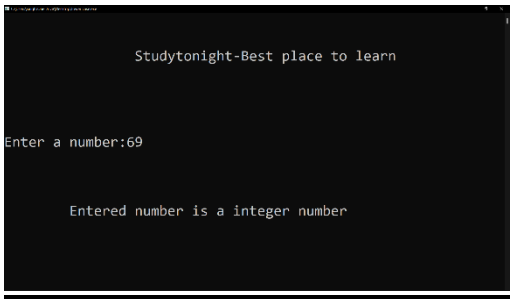


Program 4:

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
int main()
{
    printf("\n\n\t\tStudytonight-Best
place to learn\n\n\n");
    char number[10];
    int flag=0;
    int length,i=0;
    printf("\n\nEnter a number:");
    scanf("%s",number);
    length=strlen(number);
    //till string does not end
    while(number[i++]!='\0')//same as
    while(length-->0)
    {
        if(number[i]=='.')//decimal
point is present
        {
            flag=1;
            break;
        }
```

```
    }
    //if(0) is same as if(false)
    if(flag)
    printf("\n\n\n\tEntered number is a floating point number\n\n");
    else
    printf("\n\n\n\tEntered number is a integer number\n\n");
    printf("\n\n\n\n\t\t\tCoding is fun!\n\n\n");
    return 0;
}
```

Output :



Discussion and conclusion:

The program is focused on finding the output . From this lab, I understood about strings functions the and finding out different outputs of the program. Hence, the correct output was placed after each code.