Assignment-1

1. Write a program to find the length of a given string.
2. Write a program to concate two given string.
3. Write a program to copy one string to another string.
4. Write a program to compare two given string.
5. Write a program to search for the first occurrence of the character ‘c’ in the given string.
6. Write a program to find sub string is there in given string or not?
7. Write a program to generate reverse of a string.
8. Write a program to replace all vowels with star (*) and consonants with hash (#) of string.

mystring.h -

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

int length(char a[]){
int count=0,i=0;
    while(a[count]!='\0'){
        count++;
    }
    return count;
}

char * concat(char b[],char c[]){

    int count=0;
    for(int i=0;b[i]!='\0';i++){
        count++;
    }
    for(int i=0;c[i]!='\0';i++){
        b[count]=c[i];
        count++;
```

```
    }
    return b;
}

char * copy(char c[],char a[]){
    int j=0;
    char * x;
    x = (char *)calloc(sizeof(char), 100);
    for(int i=0;a[i]!='\0';i++){
       c[i]=a[i];
    }
    for(int i=0;c[i]!='\0';i++){
       x[i]=c[i];
    }
    return x;
}

int cmp(char x[],char y[]){
    int i,j=0;
    int len1,len2,max;
    len1=length(x);
    len2=length(y);
    if (len1!=len2){
        return 0;
    }
    else{
        for(i=0;i<len1;i++){
            if(x[i]==y[i]){
             j++;
            }
        }
        if(j==len1){
            return 1;
        }
        else{
            return 0;

        }
    }
```

```
}

int src (char x[], int len){                //Search 'c'
    int j, flag=0;
    for(int i=0;x[i]!='\0';i++){
        if(x[i]=='c'){
            j=i+1;
            flag=1;
            break;
        }
    }
    if(flag == 0){
        return -1;
    }
    else{
        return j;
    }
}

void reverse(char x[]){
    char temp;
    int j=length(x);
    for(int i=0;i<j/2;i++){
        temp = x[i];
        x[i]=x[j-i-1];
        x[j-i-1] = temp;
    }
    for(int m=0;x[m]!='\0';m++){
        printf("%c",x[m]);
    }
}

void replace(char x[]){
    for(int i=0;i<length(x);i++){
        if(x[i]=='a'||x[i]=='e'||x[i]=='i'||x[i]=='o'||
x[i]=='u'){
            x[i]='*';
        }else{
            x[i]='#';
```

```
        }
        printf("%c",x[i]);
    }
}

void substr(char s[],char search[]){
    int count1=0,count2=0,i,j,flag;
    count1=length(s);
    count2=length(search);
    for (i = 0; i <= count1 - count2; i++)
    {
        for (j = i; j < i + count2; j++)
        {
            flag = 1;
            if (s[j] != search[j - i])
            {
                flag = 0;
                break;
            }
        }
        if (flag == 1)
            break;
    }
    if (flag == 1)
        printf("1");
    else
        printf("0");
}

char * trans(char a[]){
    for(int i =0;a[i]!='\0';i++){
        if((a[i] == 'a')||(a[i] == 'e')||(a[i] == 'i')||(a[i]
== 'o')||(a[i] == 'u')){
            a[i] = '*';
        }
        else if(a[i] == ' '){
            a[i] = ' ';
        }
        else{
```

```
            a[i] = '#';
        }
    }
    return a;
}
```

Main-code -

```
#include "mystring.h"
#include <stdio.h>
int main(){
    char a[100], b[100], c[100];
    printf("\nEnter the string - ");
    gets(a);
    printf("\nLength of the array - %d\n", length(a));
    printf("\nEnter the 2nd String - ");
    gets(b);
    printf("\nConcatenated String - %s\n", concat(a, b));
    printf("\nCopied String: %s\n", copy(c, b));
    printf("\nComparing : %s | %s : %d\n", a, b, cmp(a,b));
    printf("\nIndex of 'c': %d\n", src(a, length(a)));
    printf("\nFinding 2nd string in 1st string String - ");
    substr(a,b);
    printf("\nReversed 1st string - ");
    reverse(a);
    printf("\n* with Vowel & # with Consonants - \n%s",
trans(a));
}
```

Output -

```
warning: this program uses gets(), which is unsafe.
Enter the string - hello good morning there

Length of the array - 24

Enter the 2nd String - good

Concatenated String - hello good morning theregood

Copied String: good

Comparing : hello good morning theregood | good : 0

Index of 'c': -1

Finding 2nd string in 1st string String - 1
Reversed 1st string - doogereht gninrom doog olleh
* with Vowel & # with Consonants -
#**#*#*## ##*##*# #**# *###*#
```