Name: Naren Chandrashekhar	SRN: PES2UG20CS216	Section: G
	Date: 3/06/2021	Week Number: 5

1

1)Write functions to

a) Reverse a string.

b) Check for equality of strings.

Input1:

Enter string

abbcbba

Output1:

Reversed string is = abbcbba

Given string is abbcbba is palindrome

Input2:

Enter string

hi

Output2:

Reversed string is = ih

Given string is hi is not palindrome

Program:

CLIENT FILE

```
#include<stdio.h>
#include<conio.h>
#include "server.h"

int main()
{
        char s[100],rev[50],r;
        printf("Enter a string ");
        scanf("%s",&s);
        reverse(s,rev);
        printf("Reveresed sting is %s\n",rev);
        if(!strcmp1(s,rev))
                printf("The string is a palindrome");
        else
                printf("The string is not a palindrome");
        return 0;
}
```

Server.h

```
void reverse(char *s, char *rev);
int strcmp1(char *s1, char *s2);
```

server.c

```
#include<string.h>

void reverse(char *s,char *rev)
{
        int i,j,len;
        len = strlen(s);
        for(i=0,j=len-1;i<len;i++,j--)
        {
                *(rev+i) = *(s+j);
        }
        rev[len] = '\0';
}
int strcmp1(char *s1, char *s2)
{
        int i,j,len;
        len = strlen(s1);
        for(i=0,j=0;i<len;i++,j++)
        {
                if(*(s1+i) == *(s2+j))
                        continue;
                else if(*(s1+i) > *(s2+j))
                        return 1;
                else
                        return -1;
        }
        return 0;
}
```

Output Screenshot:

```
D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_5\Program1>gcc -c server.c

D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_5\Program1>gcc -c Program1.c

D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_5\Program1>gcc server.o Program1.o

D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_5\Program1>a
Enter a string madam
Reveresed sting is madam
The string is a palindrome
D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_5\Program1>a
Enter a string programming
Reveresed sting is gnimmargorp
The string is not a palindrome
```

2 Write function to find all occurrences of a character in a string and use this function to replace all occurences of a character by specific character.

Input1:
Enter the string : Welcome to C programming
Enter a character to replace: o
Enter character to replace with r : @
Output1:
Before replace: Welcome to C programming
After replace: Welc@me t@ C pr@gramming

Program:

CLIENT FILE

```
#include<stdio.h>
#include<conio.h>
#include "server.h"
int main()
{
        char s[40],ch,ch1;
        printf("Enter a string ");
        scanf("%[^\n]",&s);
        fflush(stdin);
        printf("Enter a character to replace ");
        ch = getchar();
        fflush(stdin);
        printf("Enter character to replace with ");
        ch1 = getchar();
        replace(s,ch,ch1);
        printf("The modified sting is %s",s);
        return 0;
```

```
}
```

server.c

```
#include<string.h>
void replace(char *s, char ch1, char ch2)
{
        int i,len;
        len = strlen(s);
        for(i=0;i<len;i++)
        {
                if(*(s+i)==ch1)
                        *(s+i) = ch2;
        }
}
```

server.h

```
void replace(char *s, char ch1, char ch2);
```

Output Screenshot:

```
D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_5\Program2>gcc -c server.c

D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_5\Program2>gcc -c Program2.c

D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_5\Program2>gcc server.o Program2.o

D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_5\Program2>a
Enter a string welcome to C programming
Enter a character to replace o
Enter character to replace with x
The modified sting is welcxme tx C prxgramming
D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_5\Program2>a
Enter a string hello world
Enter a character to replace l
Enter character to replace with @
The modified sting is he@@o wor@d
```

3 Write a function to remove all repeated characters from a given string and display the string without duplicate characters.

Input 1:

Enter any string: hello world

Output 1:

String before removing duplicates: hello world

String after removing duplicates: helo wrd

Input 1:

Enter any string: programming in c

Output 1:

String before removing duplicates: programming in c

String after removing duplicates: progamin c

Program:

CLIENT FILE

```
#include<stdio.h>
#include<conio.h>
#include "server.h"

int main()
{
        char s[40],mod[40];
        printf("Enter a string ");
        scanf("%[^\n]",&s);
        fflush(stdin);
        printf("The string before duplication is %s\n",s);
        remduplicate(s,mod);
        printf("The string after removing duplication is %s",mod);
        return 0;
}
```

server.c

```
#include<string.h>
void remduplicate(char *s, char *remdup)
{
        int i,len,j,count,found;
        len = strlen(s);
        remdup[0] = s[0];
        count = 1;
        for(i=1;i<len;i++)
        {
                found = 0;
                for(j=0;j<=count;j++)
                {
```

```
                        if(*(s+i)==*(remdup+j))
                        {
                                found = 1;
                                break;
                        }
                }
                if(!found)
                {
                        remdup[count]=s[i];
                        //printf("%c\n",remdup[count]);
                        count++;
                }
        }
        remdup[count]='\0';
}
```

server.h

```
void remduplicate(char *s, char *mod);
```

Output Screenshot:

```
D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_5\Program3>gcc -c server.c

D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_5\Program3>gcc -c Program3.c

D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_5\Program3>gcc server.o Program3.o

D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_5\Program3>a
Enter a string hello world
The string before duplication is hello world
The string after removing duplication is helo wrd
D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_5\Program3>a
Enter a string welcome to c programming
The string before duplication is welcome to c programming
The string after removing duplication is welcom tprgain
```

4 Write function to Concatenate two strings and use this to concatenate n (i.e, say 2) strings.

Input 1:

Enter 1st string

pes

Enter 2nd string

	university Enter number of times u want to append 1 **Output1:** Concatenated string is pesuniversity **Input2:** Enter 1st string pes Enter 2nd string university Enter number of times u want to append 2 **Output2:** Concatenated string is pesuniversityuniversity

	Program:
	CLIENT FILE
	#include<stdio.h>
	#include<conio.h>
	#include "server.h"
	
	int main()
	{
	int n;
	char s1[40],s2[40];
	printf("Enter the first string ");
	scanf("%s",&s1);
	printf("Enter the second string ");

```
        scanf("%s",&s2);
        printf("Enter the number of times you want to append ");
        scanf("%d",&n);
        append(s1,s2,n);
        return 0;
}
```

server.h

```
void append(char *s1, char *s2, int n);
```

server.c

```
#include<stdio.h>
#include<string.h>
void append(char *s1, char *s2, int n)
{
        int i,len,j,k,m;
        char newstring[80];
        len = strlen(s1);
        for(i=0;i<len;i++)
                *(newstring+i) = *(s1+i);
        for(m=0;m<n;m++)
        {
                for(j=len,k=0;k<strlen(s2);j++,k++)
                        *(newstring+j) = *(s2+k);
                len=j;
        }
        newstring[j]='\0';
        printf("The concatenated string is %s",newstring);
}
```

Output Screenshot:

```
D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_5\Program4>gcc -c server.c

D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_5\Program4>gcc -c Program4.c

D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_5\Program4>gcc server.o Program4.o

D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_5\Program4>a
Enter the first string hello
Enter the second string world
Enter the number of times you want to append 3
The concatenated string is helloworldworldworld
D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_5\Program4>a
Enter the first string body
Enter the second string building
Enter the number of times you want to append 2
The concatenated string is bodybuildingbuilding
```

Practice Programs

1 Write a function to count the number of occurrences of a given character. Use this to find the number of occurrences of every character in a word.

Input:

pesit pes!

Output:

i occurs is 1 times

t occurs is 1 times

occurs is 1 times

p occurs is 2 times

e occurs is 2 times

s occurs is 2 times

! occurs is 1 times

Program:

CLIENT FILE

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

```
#include "server.h"
int main()
{
        char s[50],s1[50];
        int i,len;
        printf("Enter a string ");
        scanf("%[^\n]",&s);
        fflush(stdin);
        printf("%s",s);
        finduniquechar(s,s1);
        len = strlen(s1);
        for(i=0;i<len;i++)
        {
                printf("\n%c occurs %d times",s1[i],findoccurence(s,s1[i]));
        }
        return 0;
}
```

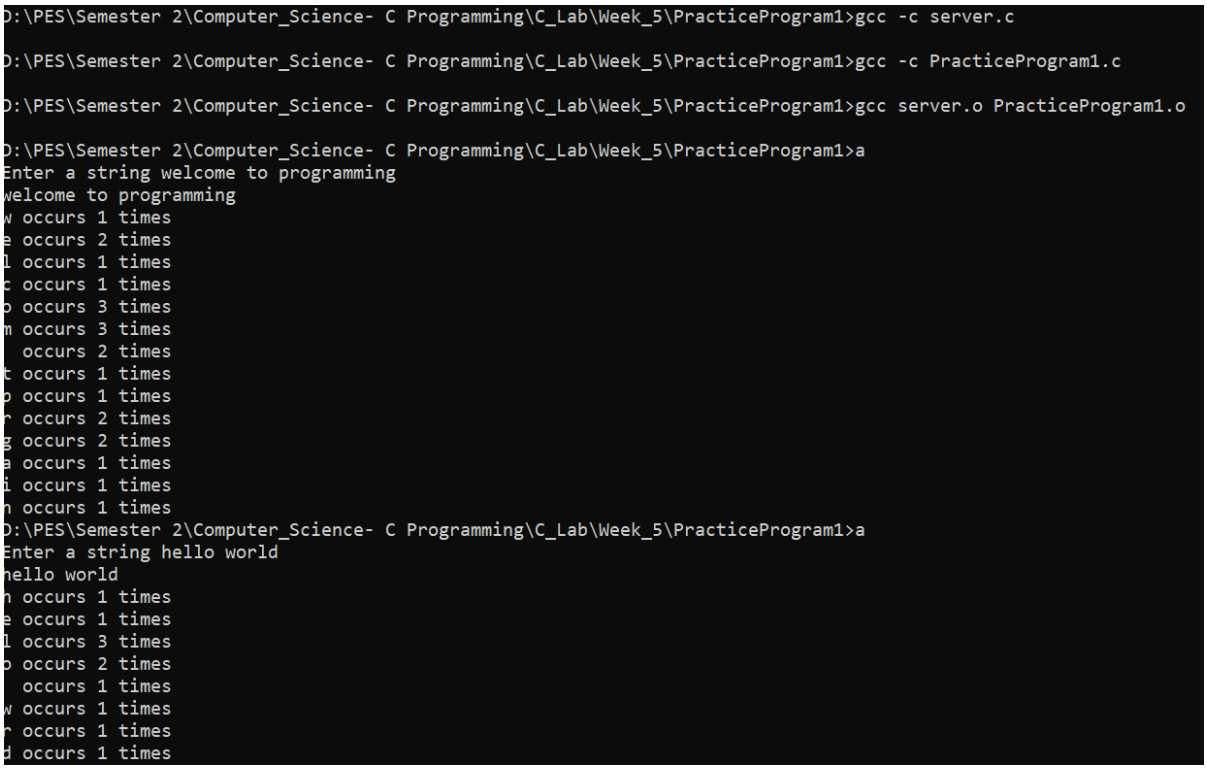
server.h

```
int findoccurence(char *s,char ch);
void finduniquechar(char *s, char *s1);
```

server.c

```
#include<string.h>
void finduniquechar(char *s, char *s1)
{
        int i,len,j,count,found;
        len = strlen(s);
        s1[0] = s[0];
        count = 1;
```

```
        for(i=1;i<len;i++)
        {
                found = 0;
                for(j=0;j<=count;j++)
                {
                        if(*(s+i)==*(s1+j))
                        {
                                found = 1;
                                break;
                        }
                }
                if(!found)
                {
                        s1[count]=s[i];
                        count++;
                }
        }
        s1[count]='\0';
}
int findoccurence(char *s, char ch1)
{
        int i,len,count=0;
        len = strlen(s);
        for(i=0;i<len;i++)
        {
                if(*(s+i)==ch1)
                        count++;
        }
```

	return count; } Output Screenshot: 
2	Write the function strend (s , t), which returns 1 if the string t occurs at the end of the string s, and zero otherwise. **Input1:** hello world! world **Output 1:** 0 **Input2:** hello world! world world **Output 2:**

	1
	Program:
	Output Screenshot: