







Name: SUNDEEP A	SRN: PES1UG20CS445	Section: O
	Date: 3-06-2021	Week Number: 5

1	<p>1) Write functions to</p> <p>a) Reverse a string.</p> <p>b) Check for equality of strings.</p> <p>Input1: Enter string abbcbbba</p> <p>Output1: Reversed string is = abbcbbba Given string is abbcbbba is palindrome</p> <p>Input2: Enter string hi</p> <p>Output2: Reversed string is = ih Given string is hi is not palindrome</p>
	<p>Program:</p> <p> client1.c - Notepad</p> <p>File Edit Format View Help</p> <pre>#include<stdio.h> #include"1.h" int main() { char s[50]; char rev[50]; printf("Enter string \n"); scanf("%s",s); reverse_string(s,rev); printf("Reversed string is = %s\n",rev); int r = check_equal(s,rev); if(r==0) printf("Given string %s is palindrome\n",s); else printf("Given string %s is not a palindrome\n",s); return 0; }</pre>

<div data-bbox="261 363 509 399"> server1.c - Notepad</div> <div data-bbox="261 407 594 438">File Edit Format View Help</div> <pre data-bbox="261 445 953 1268">#include "1.h" void reverse_string(const char*s1, char *s2) { int len = 0; while(*s1 != '\0') { ++s1; ++len; } while(len >= 0) { *s2++ = *(--s1); len--; } *s2 = '\0'; } int check_equal(char *s1, char *s2) { while(*s1 && *s2 && *s1 == *s2) { ++s1; ++s2; } return *s1 - *s2; }</pre> <div data-bbox="261 1289 459 1327"> 1.h - Notepad</div> <div data-bbox="261 1344 591 1375">File Edit Format View Help</div> <pre data-bbox="261 1381 964 1449">void reverse_string(const char*s1, char *s2); int check_equal(char *s1, char *s2);</pre> <div data-bbox="261 1484 477 1520"> 1.mk - Notepad</div> <div data-bbox="261 1537 591 1568">File Edit Format View Help</div> <pre data-bbox="261 1575 760 1778">a.exe:client1.o server1.o gcc client1.o server1.o client1.o:client1.c 1.h gcc -c client1.c server1.o:server1.c 1.h gcc -c server1.c</pre>	Output Screenshot:
---	---------------------------


	<pre> C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 5\1>mingw32-make -f 1.mk gcc -c client1.c gcc client1.o server1.o C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 5\1>a Enter string MADAM Reversed string is = MADAM Given string MADAM is palindrome C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 5\1>a Enter string computer Reversed string is = retupmoc Given string computer is not a palindrome </pre>
2	<p>Write function to find all occurrences of a character in a string and use this function to replace all occurrences of a character by specific character.</p> <p>Input1: Enter the string : Welcome to C programming Enter a character to replace: o Enter character to replace with r : @</p> <p>Output1: Before replace: Welcome to C programming After replace: Welc@me t@ C pr@gramming</p>
	<p>Program:</p> <p> client2.c - Notepad</p> <pre> File Edit Format View Help #include<stdio.h> #include<stdlib.h> #include<string.h> #include"header1.h" int main() { char str1[1000],c1,c2; printf("Enter the string : "); scanf("%[^\n]s",str1); printf("Enter a character to replace : "); fflush(stdin); scanf("%c",&c1); </pre>

```
printf("Enter character to replace with %c : ",c1);
fflush(stdin);
scanf("%c",&c2);
printf("\n Before replace :%s",str1);
replacechar(str1,c1,c2);
printf("\nAfter replace : %s",str1);
return 0;
}
```

 server2.c - Notepad


File Edit Format View Help

```
#include<stdio.h>
#include<string.h>
#include"header1.h"
void replacechar(char *s,char c1,char c2)
{
    for(int i=0;s[i];i++)
        if(s[i]==c1)
            s[i]=c2;
}
```

 header1.h - Notepad

File Edit Format View Help


```
void replacechar(char *s,char c1,char c2);
```

 2.mk - Notepad

File Edit Format View Help

```
a.exe:client2.o server2.o
gcc client2.o server2.o
client2.o:client2.c header1.h
gcc -c client2.c
server2.o:server2.c header1.h
gcc -c server2.c
```

Output Screenshot:

	<pre> C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 5\2>mingw32-make -f 2.mk gcc -c client2.c gcc client2.o server2.o C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 5\2>a Enter the string : Welcome to C Programming Enter a character to replace : o Enter character to replace with o : @ Before replace :Welcome to C Programming After replace : Welc@me t@ C Pr@gramming </pre>
3	<p>Write a function to remove all repeated characters from a given string and display the string without duplicate characters.</p> <p>Input 1:</p> <p>Enter any string: hello world</p> <p>Output 1:</p> <p>String before removing duplicates: hello world</p> <p>String after removing duplicates: helo wrd</p> <p>Input 1:</p> <p>Enter any string: programming in c</p> <p>Output 1:</p> <p>String before removing duplicates: programming in c</p> <p>String after removing duplicates: progamin c</p>
	<p>Program:</p>  client3.c - Notepad <pre> File Edit Format View Help #include<stdio.h> #include"header3.h" #define MAX_SIZE 100 int main() { char str[MAX_SIZE]; printf("Enter any string: "); scanf("%[^\n]s",str); printf("String before removingduplicates: %s\n",str); removeDuplicates(str); printf("String after removing duplicates : %s\n",str); return 0; } </pre>

server3.c - Notepad


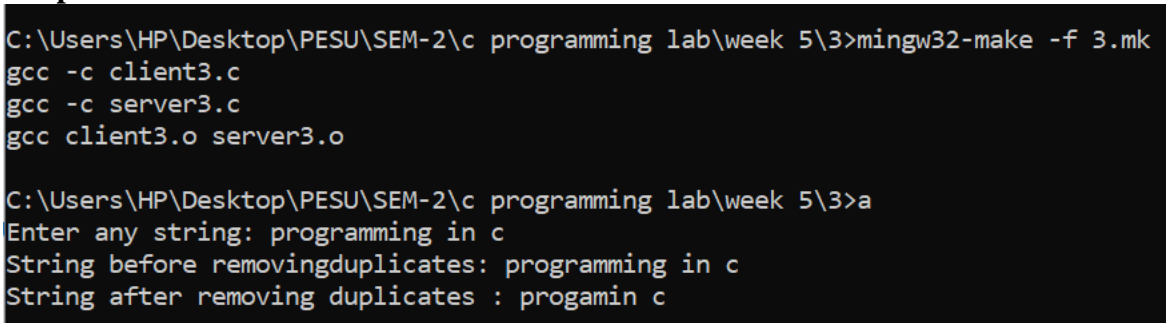
File Edit Format View Help


```
#include "header3.h"
void removeDuplicates(char *str)
{
    int i=0;
    while(str[i]!='\0')
    {
        removeAll(str, str[i], i+1);
        i++;
    }
}
void removeAll(char *str, const char toRemove, int index)
{
    int i;
    while(str[index]!='\0')
    {
        if(str[index]==toRemove)
        {
            i=index;
            while(str[i]!='\0')
            {
                str[i]=str[i+1];
                i++;
            }
        }
        else
            index++;
    }
}
```




header3.h - Notepad

File Edit Format View Help

```
void removeDuplicates(char *str);
void removeAll(char *str, const char toRemove, int index);
```

	 3.mk - Notepad File Edit Format View Help a.exe:client3.o server3.o gcc client3.o server3.o client3.o:client3.c header3.h gcc -c client3.c server3.o:server3.c header3.h gcc -c server3.c
	Output Screenshot:  <pre> C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 5\3>mingw32-make -f 3.mk gcc -c client3.c gcc -c server3.c gcc client3.o server3.o C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 5\3>a Enter any string: programming in c String before removingduplicates: programming in c String after removing duplicates : progamin c </pre>
4	<p>Write function to Concatenate two strings and use this to concatenate n (i.e, say 2) strings.</p> <p>Input 1:</p> <p>Enter 1st string</p> <p>pes</p> <p>Enter 2nd string</p> <p>university</p> <p>Enter number of times u want to append</p> <p>1</p> <p>Output1:</p> <p>Concatenated string is pesuniversity</p> <p>Input2:</p>

	<p>Enter 1st string</p> <p>pes</p> <p>Enter 2nd string</p> <p>university</p> <p>Enter number of times u want to append</p> <p>2</p> <p>Output2:</p> <p>Concatenated string is pesuniversityuniversity</p>
	<p>Program:</p> <p> client_strcat.c - Notepad</p> <p>File Edit Format View Help</p> <pre>#include<stdio.h> #include"server_strcat.h" int main() { char s1[100];char s2[50]; int n; printf("Enter 1st\n"); scanf("%s",s1); printf("Enter 2nd string\n"); scanf("%s",s2); printf("Enter number of times you want to append\n"); scanf("%d",&n); printf("Concatinated string is %s\n",my_strcat(s1,s2,n)); return 0; }</pre>

	<div data-bbox="267 363 1015 898">  server_strcat.c - Notepad File Edit Format View Help #include "server_strcat.h" void my_str_cat(char *d, const char *s) { while(*d) d++; while(*d++=*s++); } char* my_strcat(char *d1, const char *s1, int n) { for(int i=0; i<n; i++) my_str_cat(d1, s1); return d1; } </div> <div data-bbox="267 930 1015 1092">  server_strcat.h - Notepad File Edit Format View Help void my_str_cat(char *d, const char *s); char* my_strcat(char *d1, const char *s1, int n); </div> <div data-bbox="267 1123 1015 1428">  4.mk - Notepad File Edit Format View Help a.exe: client_strcat.o server_strcat.o gcc client_strcat.o server_strcat.o client_strcat.o: client_strcat.c server_strcat.h gcc -c client_strcat.c server_strcat.o: server_strcat.c server_strcat.h gcc -c server_strcat.c </div>
	Output Screenshot:

```
C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 5\4>mingw32-make -f 4.mk  
gcc -c server_strcat.c  
gcc client_strcat.o server_strcat.o
```

```
C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 5\4>a  
Enter 1st  
PES  
Enter 2nd string  
University  
Enter number of times you want to append  
2  
Concatinated string is PESUniversityUniversity
```

```
C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 5\4>a  
Enter 1st  
PES  
Enter 2nd string  
University  
Enter number of times you want to append  
1  
Concatinated string is PESUniversity
```