Lab Report-1

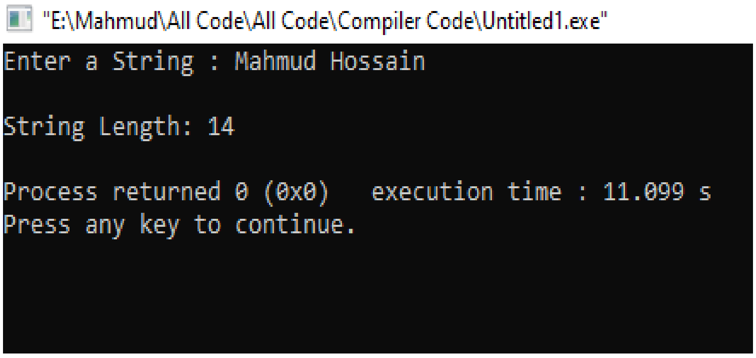
Name: Mahmud

ID: 181-15-1862

Experiment: 01
Experiment Name: Length of string in C language .
Code:

```
#include<stdio.h>
int main()
{
    char s1[20];
    int i,len=0;

    printf("Enter a String : ");
    gets(s1);

    for(i=0;s1[i]!='\0';i++){
        len++;
    }
    printf("\nString Length: %d\n",len);
    return 0;

}
```

Output:



"E:\Mahmud\All Code\All Code\Compiler Code\Untitled1.exe"

```
Enter a String : Mahmud Hossain

String Length: 14

Process returned 0 (0x0)   execution time : 11.099 s
Press any key to continue.
```

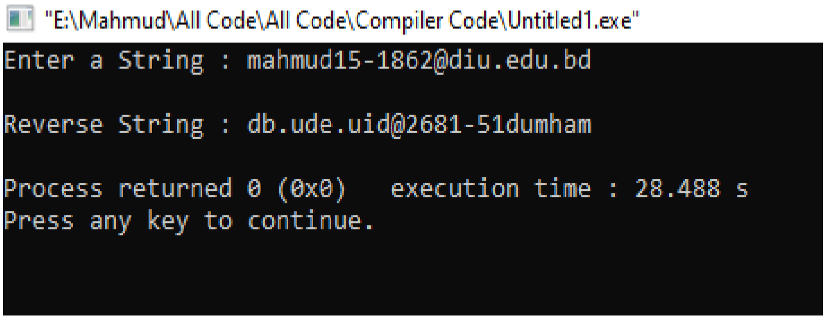
Discussion:

At first, define an array for put a string.
Use a gets function for taken user input.
Then find the length of string used for loop.

Experiment: 02
Experiment Name: **String reverse in c**
Code:

```
#include<stdio.h>
int main()
{
   char s[40],s1[40];
   int i,j,len=0;

   printf("Enter a String : ");
   gets(s);

   for(i=0;s[i]!='\0';i++){
      len++;
   }

   for(i=len-1,j=0;i>=0;i--,j++){
      s1[j]=s[i];
   }
   s1[len]='\0';

   printf("Reverse String : ");
   puts(s1);


   return 0;


}
```

Output:



"E:\Mahmud\All Code\All Code\Compiler Code\Untitled1.exe"

```
Enter a String : mahmud15-1862@diu.edu.bd

Reverse String : db.ude.uid@2681-51dumham

Process returned 0 (0x0)   execution time : 28.488 s
Press any key to continue.
```

Discussion:

At first, define two array s, s1.
Use a gets function for taken user input.
Then find the length of string used for loop.

Again use a loop for string character swap one by one from last to the first index.
Reverse string put into array s1.
At last print array s1.

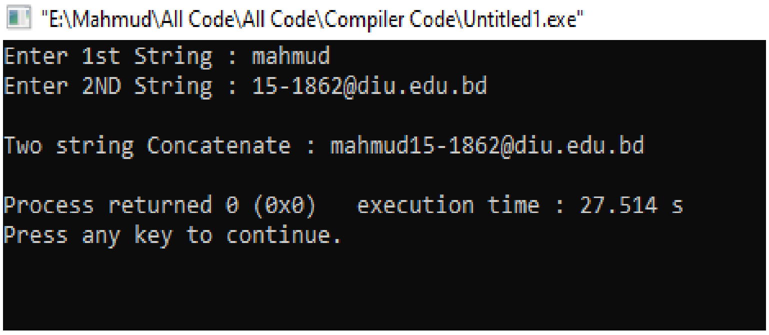
Experiment: 03

Experiment Name: Two string concatenation in c

Code:

```
#include<stdio.h>

int main()

{

   char s1[30],s2[30];

   int i,j,len=0;


   printf("Enter 1st String : ");


   gets(s1);

   printf("Enter 2ND String : ");


   gets(s2);


   for(i=0;s1[i]!='\0';i++){

      len++;

   }


   for(i=len,j=0;s2[j]!='\0';i++,j++)

   {

      s1[i]=s2[j];

   }

   s1[i]='\0';


   printf("\nTwo string Concatenate : ");

   puts(s1);
   return 0;
}
```

Output:



"E:\Mahmud\All Code\All Code\Compiler Code\Untitled1.exe"

```
Enter 1st String : mahmud
Enter 2ND String : 15-1862@diu.edu.bd

Two string Concatenate : mahmud15-1862@diu.edu.bd

Process returned 0 (0x0)   execution time : 27.514 s
Press any key to continue.
```

Discussion:

At first, define two array s1, s2.
Use a two gets function for taken user input.
Then find the length of string s1 used for loop.
Again use a loop to add second-string add with the first string.
Two strings are put into array s1.
At last print array s1.

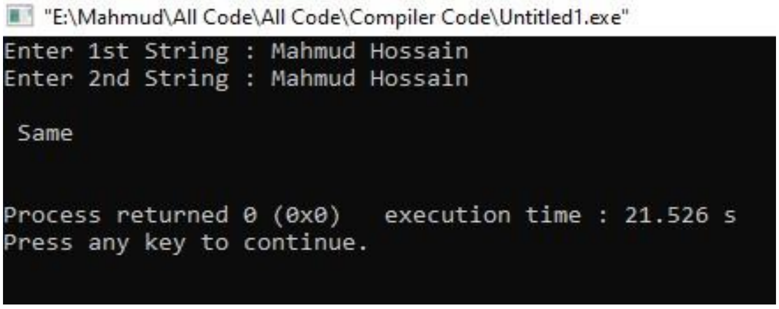
Experiment: 04

Experiment Name: Write a program to compare two strings. If the two strings are same, print "Same", otherwise print "Not Same".

Code:

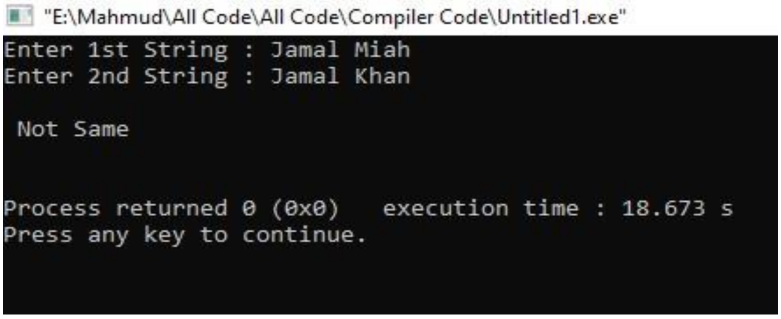
```
#include<stdio.h> int
main()
{
   char s1[20],s2[20];     int
i,j,len1=0,len2=0,cmp1=0,cmp2=0;
printf("Enter 1st String : ");     gets(s1);
printf("Enter 2nd String : ");    gets(s2);
for(i=0;s1[i]!='\0';i++){       len1++;
  }
  for(j=0;s2[j]!='\0';j++)
  {
    len2++;
  }
  if(len1!=len2){       printf("\n
Not Same \n\n");
  }else{       for(i=0,j=0;i<=len1-
1;i++,j++){
if(s1[i]==s2[j]){
cmp1++;
         }else{            printf("\n
Not Same \n\n"); break;


         }
    }
    if(len1==cmp1){
printf("\n Same \n\n");
    }
  }


  return 0;
}
```

Output



"E:\Mahmud\All Code\All Code\Compiler Code\Untitled1.exe"

```
Enter 1st String : Mahmud Hossain
Enter 2nd String : Mahmud Hossain

 Same


Process returned 0 (0x0)   execution time : 21.526 s
Press any key to continue.
```



"E:\Mahmud\All Code\All Code\Compiler Code\Untitled1.exe"

```
Enter 1st String : Jamal Miah
Enter 2nd String : Jamal Khan

 Not Same


Process returned 0 (0x0)   execution time : 18.673 s
Press any key to continue.
```

Discussion:

At first, define two array s1, s2.
Use a two gets function for taken user input.
Then find the length of both strings used for loop.
Compare string length with string one and two.
If both strings are not equal easy to tell they are not same.
Otherwise
Use a loop to compare string character by character .

Experiment: 05

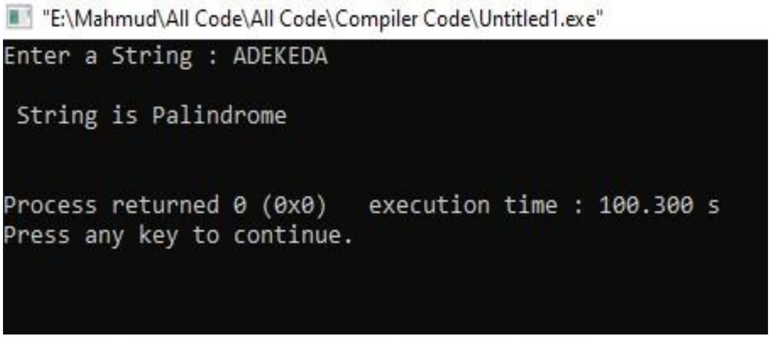
Experiment Name: Write a program to find out whether a string is a Palindrome string or not.

Code:

```
#include<stdio.h> int

main()

{
   char s1[40],s2[40];     int
i,j,len=0,len1=0,len2=0,cmp1=0,cmp2=0;


   printf("Enter a String : ");
gets(s1);


   for(i=0;s1[i]!='\0';i++){
len++;
   }
   for(i=len-1,j=0;i>=0;i--,j++){
s2[j]=s1[i];
   }
   s2[len]='\0';



   for(i=0;s1[i]!='\0';i++){
len1++;
   }


   for(j=0;s2[j]!='\0';j++)
   {
      len2++;
   }
   if(len1!=len2){        printf("\n
Not Same \n\n");
   }else{        for(i=0,j=0;i<=len1-
1;i++,j++){
if(s1[i]==s2[j]){
cmp1++;


            }else{
```

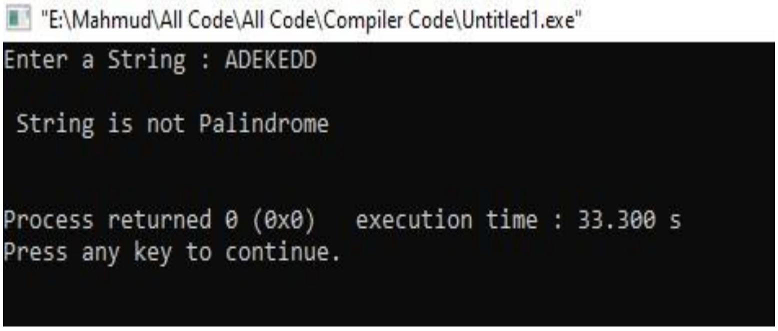
```
            printf("\n String is not Palindrome \n\n");

            break;


          }


      }

      if(len1==cmp1){


          printf("\n String is Palindrome \n\n");

      }

    }


    return 0;

}
```

Output



"E:\Mahmud\All Code\All Code\Compiler Code\Untitled1.exe"

```
Enter a String : ADEKEDA

 String is Palindrome


Process returned 0 (0x0)   execution time : 100.300 s
Press any key to continue.
```



"E:\Mahmud\All Code\All Code\Compiler Code\Untitled1.exe"

```
Enter a String : ADEKEDD

 String is not Palindrome


Process returned 0 (0x0)   execution time : 33.300 s
Press any key to continue.
```

Discussion:

At first, define two array s1, s2.
Use a gets function for taken user input.
Then find the length of strings s1 use for loop.
S1 string was reversed for the check string palindrome.
And reverse string put into S2 array.

At last compare two string s1 and reverse string s2 character by character.