Data Structures and Algorithms

Lab Report

Lab07

Group Members Name & Reg #:	**Muhammad Haris Irfan** **(FA18-BCE-090)**
Class	Data Structures and Algorithms CSC211 (**BCE-3B**)
Instructor's Name	Dilshad Sabir

In Lab Tasks

Task:1

Convert the following iterative function to a recursive one.

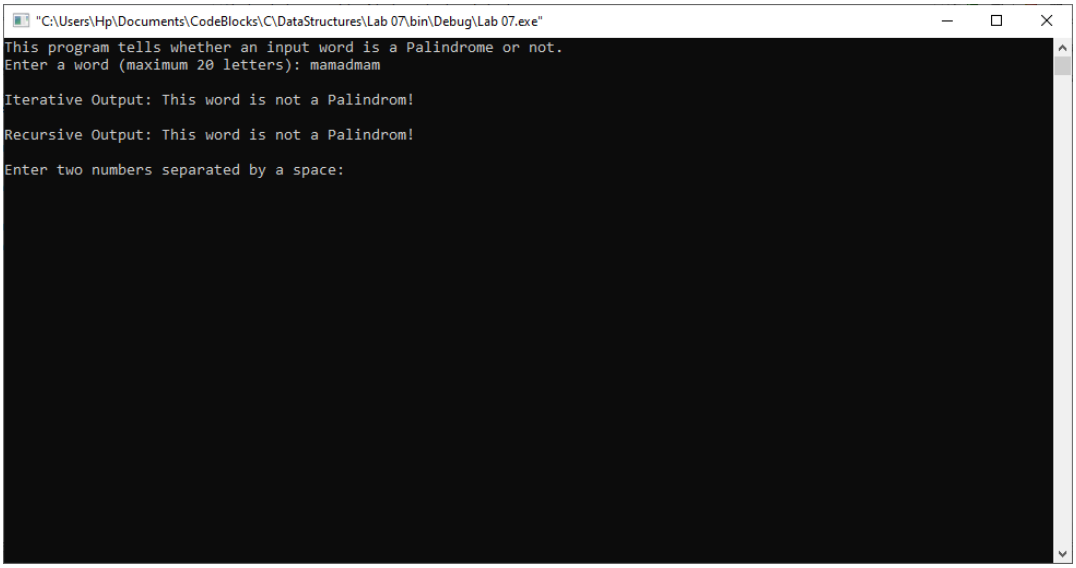
Solution:

The code is shown below,

Code:

```
bool test_palindrome_rec(char * test_word, int size)
{
    /** Complete this function **/


     if((*(test_word+iiii)) != (*(test_word+size-1)))
     {
        return(false);
     }
     else
     {

     if (size != ((firslength/2)+1))
     {
         iiii++;
          size--;
        test_palindrome_rec(test_word,size);

}     }
        return(true);

    /// You may change the prototype of the function to include more arguments.

}
```

The Result of the following code is attached below:



Task:2

Convert the following recursive function to iterative one.

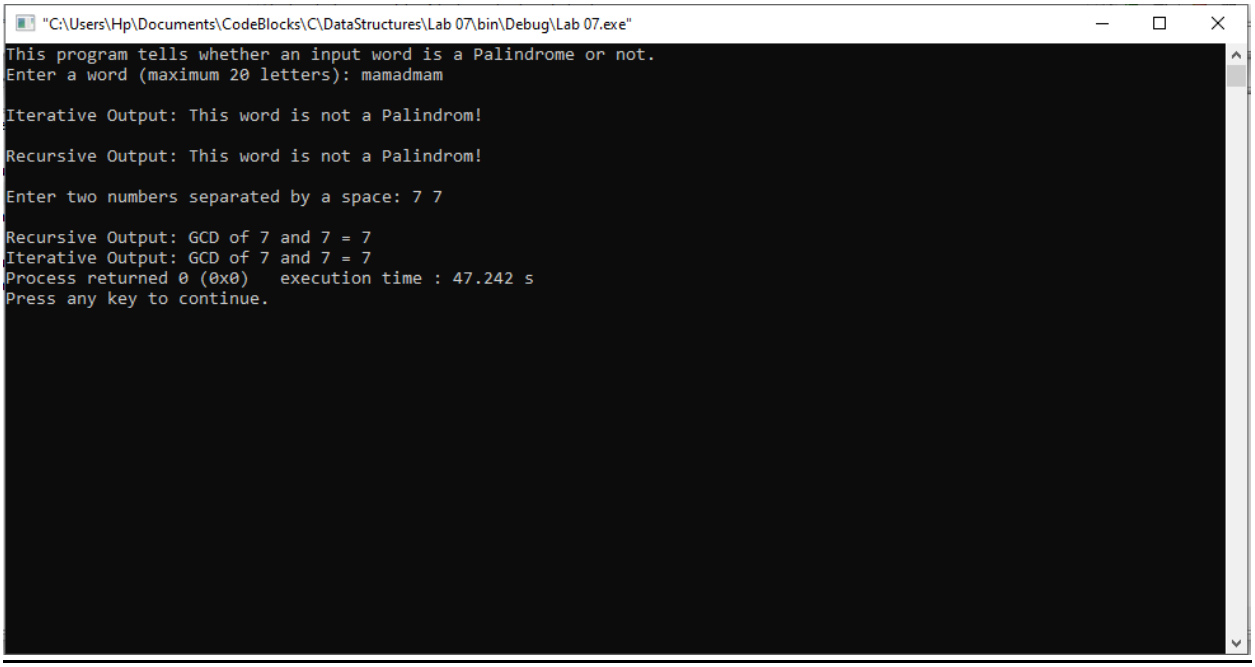
$$GCD\ (a, b) = \begin{cases} b, \text{ if } b \text{ divides } a \\ GCD\ (b, a \bmod b), \text{ otherwise} \end{cases}$$

Solution:

The code is shown below,

```
int GCD_itr(int x, int y)
{
   /** Complete this function **/
int f;
if (x<y)
{
    f=x;
    x=y;
    y=f;
}
int reml;
reml = x%y;

if(reml==0)
    return y;
else
    {
int rem2;
rem2=(x/y);
  while (rem2>1)
  {
      rem2=x/y;
      x=y;
      y=rem2;
  }
  if(rem2!=0)
return y;
    }
return x;
}
```

The Result of the following code is attached below:



Post Lab Task.

Task 3:

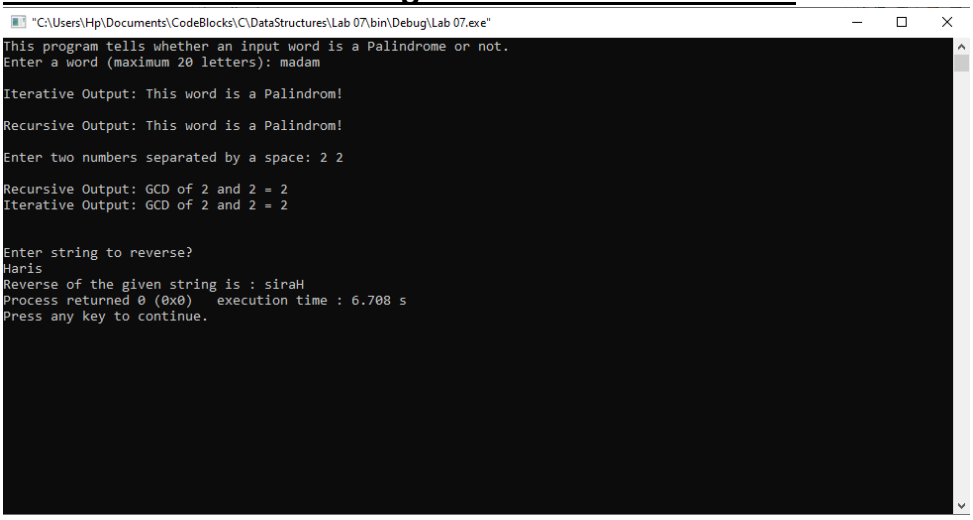
Write a program to reverse a string using recursion.

Solution

The code is shown below,

```c
void swap(char *x, char *y)
{
    char temp = *x;
    *x = *y;
    *y = temp;
}


void reverse(char str[], int l, int h)
{
    if (l < h)
    {
        swap(&str[l], &str[h]);
        reverse(str, l + 1, h - 1);
    }
}
```

The Result of the following code is attached below:



THE END