PS # 9

Deadline: 31.05.2020 at 23:50

Hand-in Policy:

Create a cpp file for each part of the assignment and save the screenshot while these parts are running.

Your assignment should consist of 4 cpp files and 4 screenshots.

Zip all files of the assignment. Your zip files should look like this

Ps9_studentnumber_name_surname.zip

1)

Write a recursive function definition for a function that has one parameter n of type int and that returns the n th Fibonacci number. The Fibonacci numbers are *F*0 is 1, *F1* is 1, *F2* is 2, *F3* is 3, *F4* is 5, and in general

Fi+2 = *Fi* + *Fi+1* for *i* = 0, 1, 2, ...

2)

Write a recursive function that has an argument that is an array of characters and two arguments that are array indexes. The function should reverse the order of those entries in the array whose indexes are between the two bounds. For example, if the array is

a[1] == 'A' a[2] == 'B' a[3] == 'C' a[4] == 'D' a[5] == 'E'

and the bounds are 2 and 5 , then after the function is run, the array elements should be

a[1] == 'A' a[2] == 'E' a[3] == 'D' a[4] == 'C' a[5] == 'B'

Embed the function in a program and test it. After you have fully debugged this function, define another function that takes a single argument that is an array that contains a string value; the function should reverse the spelling of the string value in the array argument. This function will include a call to the recursive definition you did for the first part of this project. Embed this second function in a program and test it.

3)

Write an iterative version of the recursive function in the previous project. Embed it in a program and test it.

4)

Towers of Hanoi. There is a story about Buddhist monks who are playing this puzzle with 64 stone disks. The story claims that when the monks finish moving the disks from one post to a second via the third post, time will end. Eschatology (concerns about the end of time) and theology will be left to those better qualified; our interest is limited to the recursive solution to the problem.

A stack of n disks of decreasing size is placed on one of three posts. The task is to move the disks one at a time from the first post to the second. To do this, any disk can be moved from any post to any other post, subject to the rule that you can never place a larger disk over a smaller disk. The (spare) third post is provided to make the solution possible. Your task is to write a recursive function that describes instructions for a solution to this problem. We do not have graphics available, so you should output a sequence of instructions that will solve the problem.

Hint: If you could move n-1 of the disks from the first post to the third post using the second post as a spare, the last disk could be moved from the first post to the second post. Then by using the same technique (whatever that may be), you can

move the n-1 disks from the third post to the second post, using the first post as a spare. There! You have the puzzle solved. You have only to decide what the nonrecursive case is, what the recursive case is, and when to output instructions to move the disks.