**Lab 4 – Recursion and Stacks**

**CSC 3302**

|  |  |
| --- | --- |
|  |  |
| **Introduction:** | When a function is called in a process, the current function is placed onto a Call Stack. This is a data structure the OS uses to keep the order of functions being called. When a function has finished executing it is removed from the Call Stack and the next process can run. For recursion many instances of the function are placed onto the call stack and removed upon completion. The stack is how the appropriate values are maintained despite being in the same function code as each function is treated separately. |
|  |  |
| **Description:** | For this lab you will be using a Stack ADT to simulate a recursive function call stack. You will need to solve the Fibonacci sequence for any number without simply counting the numbers with a for loop and without using recursion. Instead of looping through the numbers you must solve the recurrence below using only a single stack.   |  |  |  |  | | --- | --- | --- | --- | | f(n) |  | n = 0 | 0 | | n = 1 | 1 | | n > 1 | f(n-1) + f(n-2) | |
|  |  |
| **Program:** | You must use create a new class called **FibStack** that is an implementation of a *reference based* stack. You do not know how big the stack can get so an array will not work. You must implement **push**, **pop**, and **isEmpty** methods.  Even though Fibonacci is for integers, your stack should be generic for any datatype. Remember that generics do not work for primitives so you must use Integers for your program.  You are given a file called **p4.dat** in the folder on the server. This is a list of integers that you must solve. Display the number and the correct Fibonacci number. Remember that Fibonacci should not process any number under 0. |
|  |  |
| **Input:** | There is an input file called **p4.dat** were each line contains a single integer. You must read in this value and print out the Fibonacci value for it. |

|  |  |
| --- | --- |
| **Output:** | Fibonacci value for number: 1 is 1  Fibonacci value for number: 3 is 2  Fibonacci value for number: 5 is 5  Fibonacci value for number: 6 is 8  Fibonacci value for number: 9 is 34  Fibonacci value for number: 7 is 13 |

|  |  |
| --- | --- |
|  |  |
| **Hints:** | Use appropriate software design techniques, and implement the class methods with Java constructs for I/O, declarations, and calculations.  Build your program in steps (i.e., get the input and output working, then add the functions, etc.). Emphasize functionality first, then add the advanced features. Work on this program over time, DO NOT wait until the last minute (i.e., the day before it is due) to start! |
|  |  |
| **Data:** | On the server, you will need to use the following absolute path and append the filename to it. "/home/courses/csci3302-002/datafiles/" |
|  |  |
|  | Remember that you must pass the data file name in as a command line argument. |
|  |  |
|  |  |