University of Maryland University College

CMIS 242 6382 Intermediate Programming (2165)

Submit Files - Project 3

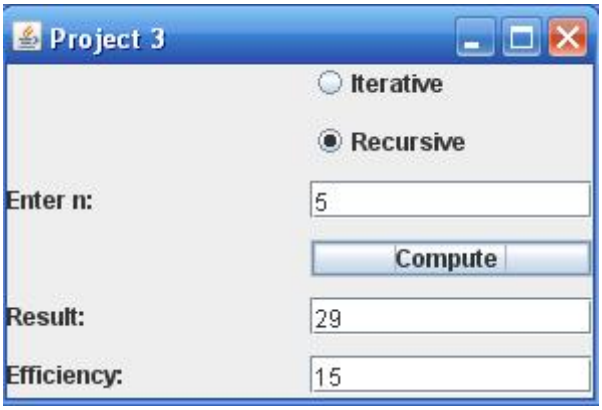
CMIS 242 6382 Intermed... Justin Casteel

Submission Folder

Project 3

Instructions

The third programming project involves writing a program to calculate the terms of the following sequence of numbers: 0 1 2 5 12 29 ... where each term of the sequence is twice the previous term plus the second previous term. The 0^{th} term of the sequence is 0 and the 1^{st} term of the sequence is 1. The interface to the program should be a GUI that looks similar to the following:



The pair of radio buttons allows the user to choose whether an iterative or recursive method is used to compute the term of the sequence. When the user enters a value for *n* and then clicks the *Compute* button, the n^{th} term of the sequence should be displayed in the *Result* field. The *Efficiency* field should contain the number of calls to the recursive method when the recursive option is chosen and the number of iterations of the loop when the iterative option is selected.

The *Iterative* radio button should be initially set to selected.

When the window is closed, the efficiency values should be computed with values of *n* from 0 to 10 and written to a file. Each line of the file should contain the value of *n*, the efficiency of the iterative method for that value of *n* and the efficiency of the recursive method. The values should be separated by commas so the file can be opened with Excel and used to graph the value of the efficiencies for both the iterative and recursive options along the *y* axis with the value of *n* along the *x*-axis. The graph should be included as your test plan in the Word document that accompanies this project and should also contain a brief explanation of the observed results.

The program should consist of two classes. The first class should define the GUI and should be hand-coded and not generated by a GUI generator. In addition to the main method and a constructor to build the GUI, an event handler will be needed to handle the *Compute* button click and another handler will be needed to produce the file described above when the window is closed. The latter handler should be an object of an inner class that extends the `WindowAdapter` class.

The other class should be named `Sequence`. It should be a utility class meaning that all its methods must be class (static) methods and no objects should be able to be generated for that class. It should contain three public methods:

1. The first method `computeIterative` should accept a value of *n* and return the corresponds element in the sequence using iteration.
2. The second method `computeRecursive` should accept a value of *n* and return the corresponds element in the sequence using recursion. This method with be a helper method because it will need to initialize the efficiency counter before calling the private recursive method that will actually perform the recursive computation.
3. The third method `getEfficiency` will return the efficiency counter left behind by the previous call to either of the above two methods.

Be sure that all instance and class variables are declared as *private*. Also any exceptions thrown by nonnumeric inputs should

Submit Cancel

CMIS 242 6382 Intermed... Justin Casteel

Submit Files

Files to submit *

(0) file(s) to submit

After uploading, you must click Submit to complete the submission.

Add a File Record Audio

Comments

Paragrap ▼