JAVA PROGRAMMING COURSE (SWE2023)

FALL SEMESTER 2023

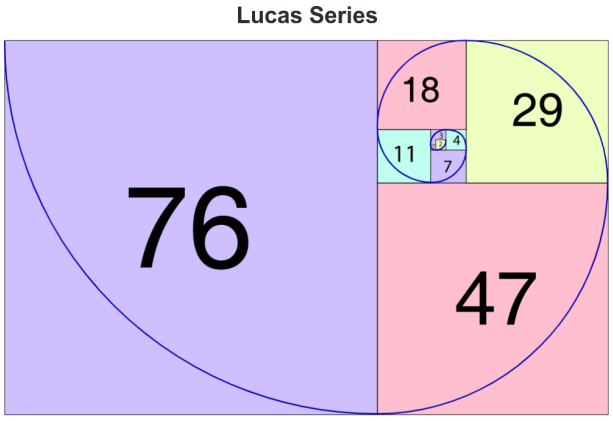
INSTRUCTOR: Prof. TAMER ABUHMED
COLLEGE OF SOFTWARE

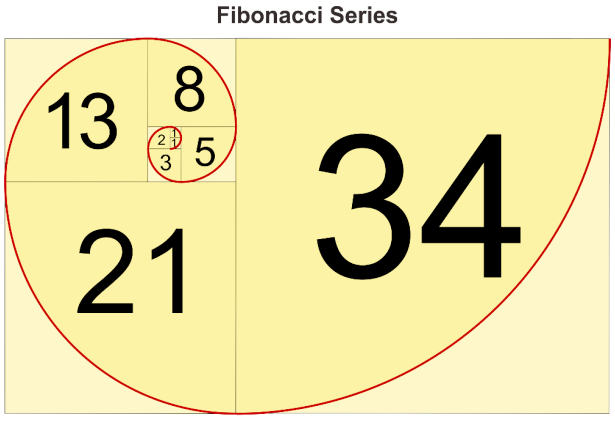
Assignment 4

This assignment consists of 1 task. Guidelines for submission format are given at the end of the assignment file.

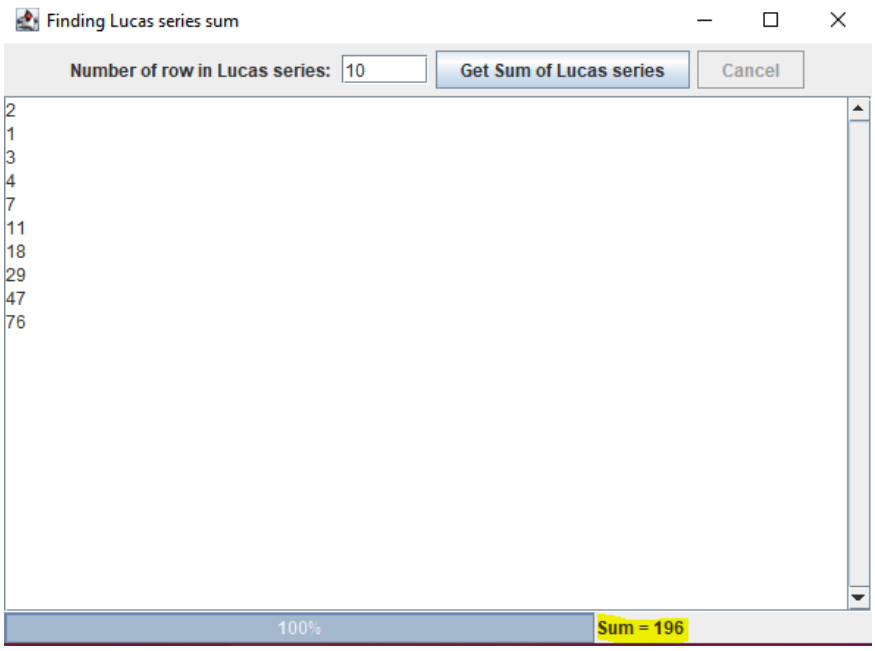
Task description:

We all know about the Fibonacci series. In **the Fibonacci series**, each number is the sum of the two previous numbers as shown. But, have you ever heard about the Lucas series? The Lucas series has the same recursive relationship as the Fibonacci sequence, where each term is the sum of the two previous terms, but with different starting values. As scientists mention the Lucas series is more close to the golden ratio as compared to the Fibonacci series. Let's make a program for this new interesting type of series!

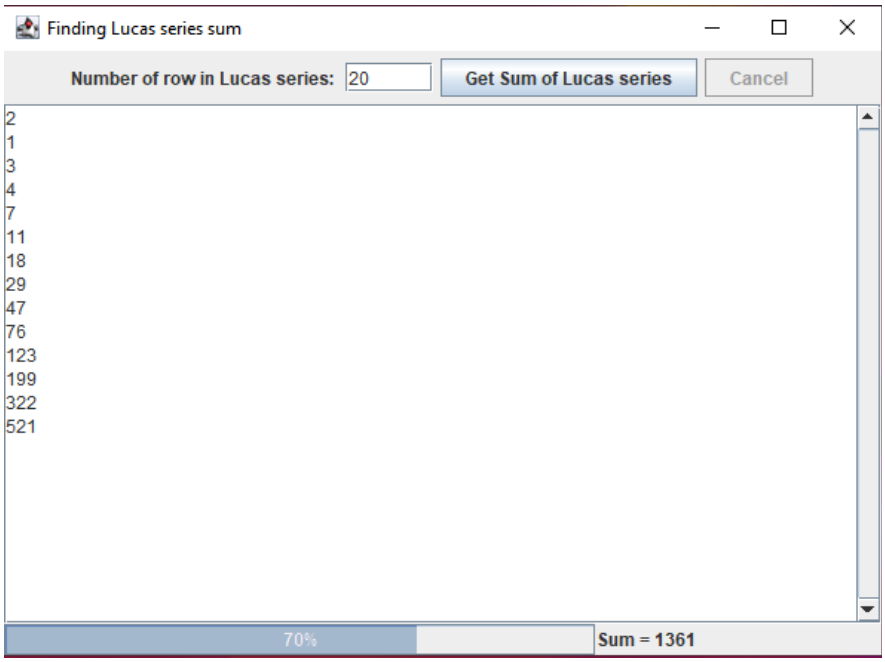




Develop a java application that calculates the sum of the Lucas series. Your program has to print each row of the Lucas series separately and print them as shown in the image below. At the same time, using another thread (use *SwingWorker*), your program has to calculate the overall sum of all rows (sum of numbers in the Lucas series). In the end, your program has to save the Lucas series into a file (as shown in the video).

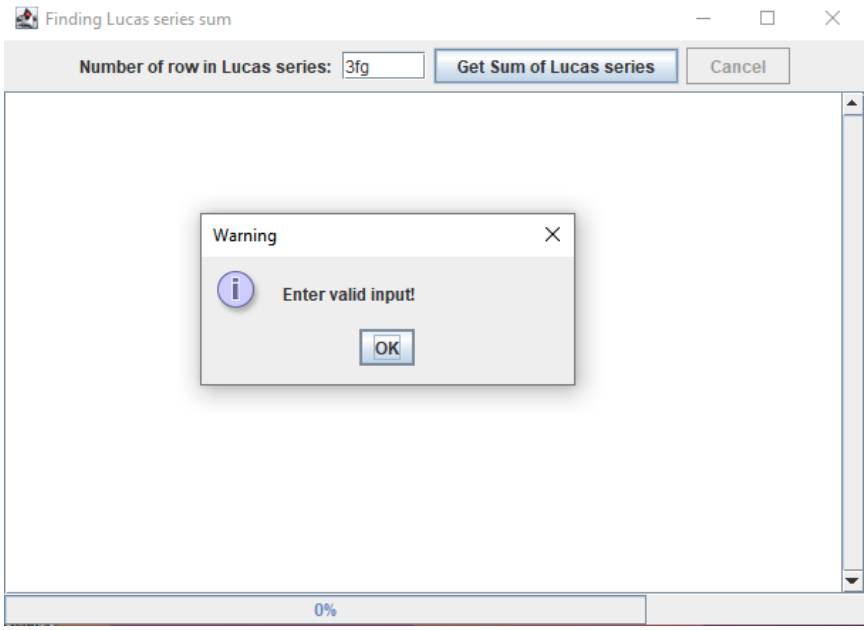


While processing, the user can press the 'cancel' button. In that case, the program should stop right there and print the sum of the Lucas series that is calculated so far. Before pressing the "Get Sum of Lucas series" button, the cancel button must be disabled.



Note: after the program ends, the cancel button should be disabled.

Exception Handling: If the user enters the wrong inputs (string, too big integers, float), show a dialog box about the error:



File I/O: Write your Lucas series into the file (.txt).

Note: during the grading, we will give test inputs between 3 and 20 **(3 <= N <= 20)**.

Requirements:

1. GUI
2. Exception handling
3. File I/O
4. Multithreading
5. Good coding style (like naming variables with meaningful names, dividing code into one-task functions)
6. Comments (meaningful comments) - "// this is variable" is not good comment
7. Recorded video (just run and show how it works with all cases, no need for code explanation)

Submission format: Submit **your project folder and video as one zip folder.**

Name of zip file: {student ID}_{Student name}_assignment4.zip

Example: 2020712837_Frank_Thomas_assignment4.zip

Important: Plagiarism is strictly prohibited. If there is any plagiarism found in the code, you will be given an "F" for the assignment evaluation.

If you have any questions about assignment 4, you can ask in the discussion section of the week or contact the TAs directly.

Good luck!