CSE 1322L - Lab 6

Introduction

In this lab, you will write an interface which requires implementing classes to be able to calculate the Fibonacci sequence up to some given number. You will then write two classes, both of which implement the interface, but that do so in different ways: one will do so iteratively (i.e.: using loops) and the other using Binet's formula.

Fibonacci is an interesting sequence. The first number of the sequence is 1, and the second number is also 1. Every subsequent number is the sum of the two previous numbers. Mathematically speaking, the sequence can be described as such:

$$F(n) = F(n-1) + F(n-2)$$

When computed, the sequence would look like this:

1, 1, 2, 3, 5, 8, 13, 21, 34, 55, …

Hence, Fibonacci(10) is 55, Fibonacci(6) is 8, etc.

Requirements

The features described below must be in your program:

- An interface FindFib, with a single method: int calculateFib(int)
- A class FibFormula, which implements FindFib
    - FibFormula must implement calculateFib() using Binet's Formula, which you can find below:

$$\frac{GoldenRatio^n - GoldenRatioConjugate^n}{\sqrt{5}}$$

where "n" is the nth element in the Fibonacci sequence and

$$GoldenRatio = \frac{1 + \sqrt{5}}{2}$$

$$GoldenRatioConjugate = \frac{1 - \sqrt{5}}{2}$$

- A class FibIteration, which implements FindFib
    - FibIteration must implement calculateFib() using loops (FOR, WHILE, or DO-WHILE)
- Driver class:
    - Creates one object of each of the above classes
    - Prompts the user for a number (you can assume the user will only enter positive integers that a smaller than 40)
    - Pass that number to calculateFib() of both objects, printing both results

Deliverables

- Lab6.java (Driver)
- FindFib.java
- FibIteration.java
- FibFormula.java
- uml.pdf (UML diagram)

Considerations

- You can calculate exponentiation and the square root by using the methods available in the Math class. Their documentation can be found below:
    - https://docs.oracle.com/javase/8/docs/api/java/lang/Math.html#sqrt-double-
    - https://docs.oracle.com/javase/8/docs/api/java/lang/Math.html#pow-double-double-
- Note that both Math methods return double, but calculateFib() must return an integer. You will have to cast your final result from a double to an int before returning from the method.
- The highest number in the sequence your program will be able to compute is the 46th number. This is because the next number on the sequence, 2971215073, is larger than what an int is capable of storing.

Sample Output (user input in red)

```
Find which position in the Fibonacci Sequence? 22
Fib of 22 using iteration is 17711
Fib of 22 using Binet's Formula is 17711
Program complete.
```

Sample Output (user input in red)

```
Find which position in the Fibonacci Sequence? 37
Fib of 37 using iteration is 24157817
Fib of 37 using Binet's Formula is 24157817
Program complete.
```