CmpSci 187 Discussion #6: Fibonacci Numbers
Pair Response Sheet

Marc Liberatore and John Ridgway

2 March 2015

The exercises today concern the Fibonacci function and a class `Fibonacci` that you are given to calculate it.

Exercise 1: Use this program to find $F(10)$, $F(15)$, and $F(20)$.

Exercise 2: Explain why this code is correct, using DJW's three questions for a recursive method

1. Does it have a base case, and is the output correct there?

2. Does it always make progress toward the base case, given its precondition?

3. If all recursive calls return the correct output, must it also return the correct output?

Exercise 3: For large enough n, this program will return an incorrect output due to integer overflow. Try to exhibit this overflow behavior behavior, but don't let your program run for more than one minute.What is the largest n for which `Fibonacci` will compute $F(n)$ in less than one minute?

Exercise 4: Now write a new non-recursive `fastFib` method (in the Fibonacci class) that runs faster than `fib`. (One way to do this is to make an array for the previous values of `fastFib(i)`, and look up those values rather than recalculate them as `fib` does. Another is to remember only the last two values as you successively calculate $F(2)$, $F(3)$, ... , all the way to $F(n)$). You need only write down the `fastFib` method here.

Exercise 5: With this new method, find the first n that causes integer overflow in the calculation, if you have not found it already.

Exercise 6: Write another new method `longFastFib`, similar to `fastFib` but that uses `long` variables in place of `int` variables to avoid the integer overflow problem. Can you demonstrate `long` overflow? If so, what is the first n for which this occurs? You need only write down the `longFastFib` method here.