

## 1. PROGRAM 8 INSTRUCTIONS, CS102, PROF. LOFTIN

The Fibonacci numbers  $F_0, F_1, F_2, \dots$  are defined recursively by

$$F_n = \begin{cases} 0 & \text{if } n = 0 \\ 1 & \text{if } n = 1 \\ F_{n-2} + F_{n-1} & \text{if } n \geq 2 \end{cases}$$

The first few are

$$0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, \dots$$

This project is to compute Fibonacci numbers in two ways: once by a straightforward application of the recursive algorithm and secondly by the recursive algorithm augmented by a process of memoization, in which the values that have already been computed are stored and then looked up rather than recalculated.

You should write a class `Fibonacci` which has a `main` method and also the following public methods:

```
public static int fib(int n)
public static int fibm(int n)
```

More specifically, you should

- Write a method `fib` that does performs the unmemoized recursion. Also count the number of times this method is called (use a global `static` variable for this).
- Create a global `static` array `F` of 100 `ints`. If `F[n]` is zero,  $F_n$  has not yet been calculated. Otherwise, `F[n]` should hold the value of  $F_n$ .
- Write a method `fibm`. If `fibm(n)` is called, the function should
  - See if  $F_n$  has already been calculated. If so, return that value.
  - Use the recursive definition (using `fibm` to calculate  $F_n$ ).
  - Store  $F_n$  in `F`.
  - Return  $F_n$ .

Once again, count the number of times the function is called (using another global `static` variable).

## 2. INPUT AND OUTPUT

Use a `Scanner` to input a positive integer. The call both `fib` and `fibm` and print out the Fibonacci number computed, together with the number of method calls needed in each case.

Here is the output of my program if the input is 30:

```
The Fibonacci number is 832040, and fib was called 2692537 times.
The Fibonacci number is 832040, and fibm was called 59 times.
```

3. DUE DATE: NOVEMBER 16, 2009

4. HOW TO TURN IT IN

Send your program `Fibonacci.java` as an email attachment to `loftin@rutgers.edu`.