

```
/**
 * UseFibonacci.java
 * @author Hawk Weisman
 * Modified from a program by Professor Gregory Kapfhammer
 *
 * PLEDGE:
 */

import com.clarkware.Profiler;
import java.io.*;

public class UseFibonacci
{

    public static void main(String[] args)
    {

        System.out.println("Begin experiment with different Fibonacci " +
            "implementations ...");
        System.out.println();

        // extract the value that was passed on the command
        // line; this is the nth fibonacci number that
        // we must calculate in the three different fashions
        Integer Num = new Integer(args[0]);
        int num = Num.intValue();

        // determine which algorithm and data type we are supposed to benchmark
        String chosenAlgorithm = args[1];
        String chosenDataType = args[2];

        if( chosenAlgorithm.equals("recursive") ||
            chosenAlgorithm.equals("all") )
        {
            if( chosenDataType.equals("int") ||
                chosenDataType.equals("all") )
            {
                // 1. RECURSIVE fibonacci (int)
                Profiler.begin("RecursiveFibonacciInt");
                int recursiveFib = RecursiveFibonacci.fib(num);
                Profiler.end("RecursiveFibonacciInt");

                System.out.println("(Recursive/int) The " + num +
                    "th Fibonacci " +
                    "number = " + recursiveFib + ".");
            }

            if( chosenDataType.equals("long") ||
                chosenDataType.equals("all") )
            {
                // 1. RECURSIVE fibonacci (long)
                Profiler.begin("RecursiveFibonacciLong");
                long recursiveFibLong = RecursiveFibonacci.fibLong(num);
                Profiler.end("RecursiveFibonacciLong");

                System.out.println("(Recursive/long) The " + num +
```

```
        "th Fibonacci " +
        "number = " + recursiveFibLong + ".");
    }
}

if( chosenAlgorithm.equals("iterative") ||
    chosenAlgorithm.equals("all") )
{
    if( chosenDataType.equals("int") ||
        chosenDataType.equals("all") )
    {
        // 2. ITERATIVE fibonacci (int)
        Profiler.begin("IterativeFibonacciInt");
        int iterativeFib = IterativeFibonacci.fib(num);
        Profiler.end("IterativeFibonacciInt");

        System.out.println("(Iterative/int) The " + num +
            "th Fibonacci " +
            "number = " + iterativeFib + ".");
    }

    if( chosenDataType.equals("long") ||
        chosenDataType.equals("all") )
    {
        // 2. ITERATIVE fibonacci (long)
        Profiler.begin("IterativeFibonacciLong");
        long iterativeFibLong = IterativeFibonacci.fibLong(num);
        Profiler.end("IterativeFibonacciLong");

        System.out.println("(Iterative/long) The " + num +
            "th Fibonacci " +
            "number = " + iterativeFibLong + ".");
    }
}

System.out.println();
Profiler.print(new PrintWriter(System.out));

System.out.println("... End experiment with different Fibonacci " +
    "implementations");
}
}
```