~/Documents/berry/23fall/csc121/eclipse-workspace/csc320-fib-dynprog/src/Fib.java

```java

abstract class Fib {
    /*
     * produces the n'th fibonacci number
     */
    public abstract long fib(int n);

    /* for profiling */

    private static long count = 0;
    public static void resetCount() { count = 0; }
    public static void incrCount() { count++; }
    public static long getCount() { return count; }
}


/* 1 */

class FibRec extends Fib {
    public long fib(int n) {
        incrCount();
        if (n <= 0) { return 0; }
        else if (n == 1) { return 1; }
        else { return fib(n-1) + fib(n-2); }
    }
}


/* 2 */

class FibMemo extends Fib {
    private long M[] = {};

    public long fib(int n) {
        if (M.length <= n) {
            M = new long[n+1];
        }
        return memoFib(n);
    }

    // memoized
    private long memoFib(int n) {
        incrCount();
        if (n <= 0) { return 0; }
        else if (n == 1) { return 1; }
        else {
            if (M[n] == 0) { // hasn't been filled in
                M[n] = memoFib(n-1) + memoFib(n-2);
            }
            return M[n];
        }
    }
}
```

```java


/* 3 */

class FibDP extends Fib {
    private long M[] = {};

    public long fib(int n) {
        if (M.length <= n) {
            M = new long[n+1];

            M[0] = 0;
            M[1] = 1;
            for (int i = 2; i <= n; i++) {
                M[i] = M[i-1] + M[i-2];
            }
        }

        return M[n];
    }
}


/* 4 */

class FibIter extends Fib {
    public long fib(int n) {
        long M2 = 0;   // i-2
        long M1 = 1;   // i-1

        for (int i = 0; i < n; i++) {
            long sum = M1 + M2;
            M2 = M1;
            M1 = sum;
        }

        return M2;
    }
}

```