Homework Turnin

Account:	6G_06 (rgalanos@fcps.edu)
Section:	6G
Course:	TJHSST APCS 2016–17
Assignment:	08–05

Receipt ID:	15eab4a5227c321d66b8bf424f0b2149

Turnin Successful!

The following file(s) were received:

Fib.java   (2813 bytes)

```java
1.  //Name:        Date:
2.  import java.util.*;
3.
4.  public class Fib
5.  {
6.      public static final int DEFAULT = 42;
7.
8.      public static void main(String[] args)
9.      {
10.         int n = DEFAULT;
11.         System.out.println("Recursive");
12.         calculate(new Fib1(), n);
13.         System.out.println("Iterative, stored in an array");
14.         calculate(new Fib2(), n);
15.         System.out.println("Recursive, stored in an arrayList");
16.         calculate(new Fib3(), n);
17.         System.out.println("Recursive, stored in a hashMap");
18.         calculate(new Fib4(), n);
19.     }
20.
21.     public static void calculate(Fibber fibber, int n)
22.     {
23.         long start = System.nanoTime();
24.         int f = fibber.fib(n);
25.         long finish = System.nanoTime();
26.         long time = finish - start;
27.
28.         System.out.print("fib(" + n + ") = " + f);
29.         System.out.println(" (" + time + " nanoseconds)");
30.         System.out.println();
31.     }
32.
33.     private static class Fib1 implements Fibber
34.     {
35.         public int fib(int n)
36.         {
37.             if(n == 1 || n == 2)
38.                 return 1;
39.             else
40.                 return fib(n - 1) + fib(n - 2);
41.         }
42.     }
43.     private static class Fib2 implements Fibber
44.     {
45.         public int fib(int n)
46.         {
47.             int[] array = new int[n+1];
48.             for(int i=1;i<array.length;i++)
49.             {
50.                 if(i==1||i==2)
```

```java
51.              array[i]=1;
52.            else
53.              array[i] = array[i-1] + array[i-2];
54.          }
55.        return array[n];
56.      }
57.    }
58.    private static class Fib3 implements Fibber
59.    {
60.      ArrayList<Integer> array;
61.      public Fib3()
62.      {
63.        array = new ArrayList<Integer>();
64.        array.add(1);
65.        array.add(1);
66.      }
67.      public int fib(int n)
68.      {
69.        if(array.size()>=n)
70.          return array.get(n-1);
71.        else
72.        {
73.          int temp = fib(n-1) + fib(n-2);
74.          array.add(temp);
75.          return temp;
76.        }
77.      }
78.    }
79.    private static class Fib4 implements Fibber
80.    {
81.      Map<Integer, Integer> map;
82.      public Fib4()
83.      {
84.        map = new HashMap<Integer, Integer>();
85.        map.put(1, 1);
86.        map.put(2, 1);
87.      }
88.      public int fib(int n)
89.      {
90.        if(map.containsKey(n))
91.          return map.get(n);
92.        else
93.        {
94.          int temp = fib(n-1) + fib(n-2);
95.          map.put(n, temp);
96.          return temp;
97.        }
98.      }
99.    }
100.
101.    private interface Fibber
102.    {
103.      public abstract int fib(int n);
104.    }
105. }
106.      /*
107.    Recursive
108.    fib(42) = 267914296 (3276558048 nanoseconds)
109.
110.    Iterative, stored in an array
111.    fib(42) = 267914296 (4988 nanoseconds)
112.
113.    Recursive, stored in an arrayList
114.    fib(42) = 267914296 (64025 nanoseconds)
115.
116.    Recursive, stored in a hashMap
117.    fib(42) = 267914296 (177793 nanoseconds)
118.
119.      */
120.
```