**Assignment 5**

**RDC 11/15/2017**

**Name:\_\_\_\_\_\_\_\_\_\_\_\_ Class:\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**( ) 1. Refer to method mystery:**

public int mystery(int n, int a, int d){

if (n == 1)

return a;

else

return d + mystery(n - 1, a, d);

}

**What value is returned by the call mystery(3, 2, 6)?**

**6+6+2**

(A) 20

(B) 14

(C) 10

(D) 8

(E) 2

**( ) 2. Consider the following method:**

public void doSomething(int n){

if (n > 0)

{

doSomething(n - 1);

System.out.print(n);

doSomething(n - 1);

}

}

**What would be the output following the call doSomething(3)?**

(A) 3211211

(B) 1121213

(C) 1213121

(D) 1211213

(E) 1123211

**( ) 3. Refer to the following recursive method.**

public int mystery(int n){

if (n < 0)

return 2;

else

return mystery(n - 1) + mystery(n - 3);

}

**What value is returned by the call mystery(3)?**

(A) 12

(B) 10

(C) 8

(D) 6

(E) 4

**( ) 4. This question refers to methods f1 and f2 that are in the same class:**

public int f1(int a, int b){

if (a == b)

return b;

else

return a + f2(a - 1, b);

}

public int f2(int p, int q){

if (p < q)

return p + q;

else

return p + f1(p - 2, q);

}

**What value will be returned by a call to f1(5, 3)?**

**5+4+2+1+3**

(A) 5

(B) 6

(C) 7

(D) 12

(E) 15

**5. [Optional]**阅读 **<http://blog.csdn.net/wangnanwlw/article/details/52460119>** 中的内容，复习上节课学的知识。

**6. (1)** 阅读 **<https://en.wikipedia.org/wiki/Fibonacci_number>** 中的内容；

**(2)** 用递归写一个程序并把你的代码粘贴在下面；

**(3)** 写好这段代码之后，尝试输出斐波那契数列的前**50**项并观察程序运行所需时间。实例视频在压缩包内。

**(4) [Optional]**尝试用其他方法，在更短时间内输出同样的结果。

**<http://cmjcmj8080.iteye.com/blog/1021340>**

import java.util.Scanner;

public class Fibonacci {

public static void main(String[] args) {

System.out.println("请输入要输出多少项斐波那契数列：");

Scanner sc = new Scanner(System.in);

int num = sc.nextInt(); //num是要输出的项数

sc.close();

for (int i = 1; i < num+1; i++){

System.out.print(fibonacci(i) + " ");

}

}

public static int fibonacci(int input){

if (input == 0||input == 1){

return input;

}else{

return fibonacci(input - 1) + fibonacci(input - 2);

}

}

}