Homework 1 -- Recursion

Basic Questions (Required):

1.

a. What does a method have to do for it to be considered recursive?

**Answer**: A method has the repeat the calculation within a smaller part of itself.

b. What is a base case, and why should every recursive method have one?

**Answer:** It is the function’s stopping point, which is necessary so that the recursion doesn’t go infinite.

c. Can every recursive method be written iteratively (without recursion)?

**Answer:** Yes.

d. What are the benefits of writing iterative code compared to recursive code?

**Answer:** Iterative code takes less memory, and is faster than recursion due to recursion’s overhead of maintaining call stack.

e. What are the benefits of writing recursive code compared to iterative code?

**Answer:** Recursion keeps your code shorter, as it only 1 base condition which is the terminate condition.

2.

a. Create a recursive method that takes an integer n and returns the sum of

the integers from 1 to n.

**Answer:**

import java.io.\*;

class sum\_of\_digits

{

static int sum\_of\_digit(int n)

{

if (n == 0)

return 0;

return (n % 10 + sum\_of\_digit(n / 10));

}

public static void main(String args[])

{

int num = 12345;

int result = sum\_of\_digit(num);

System.out.println("Sum of digits in " +

num + " is " + result);

}

}

b. Determine the return value when 7 is passed into the method,

and show all work (don't just do 1 + 2 + 3 + 4 + 5 + 6 + 7).

**Answer: 28.** For my working, I just put 1234567 into the code above and ran it in an IDE.

c. Determine what happens when -1 is passed into the method, and show all work.

**Answer: -2.** For my working, I put -101 into the code just like in the previous question.

3. Determine the output:

public class Main

{

public static void main(String[] args)

{

p3(5);

}

public static void p3(int n)

{

if (n <= 0) {System.out.println(0);}

else

{

System.out.println(n);

p3(n - 1);

}

}

}

**output**:

5

4

3

2

1

0

4. Determine why the following code causes an error:

public class Main

{

public static void main(String[] args)

{

p3(5);

}

public static void p3(int n)

{

if (n <= 0) {System.out.println(0);}

else

{

p3(n - 1);

System.out.println(n);

p3(n);

}

}

}

**Answer:** The following code causes error because of stack overflow, as there is a loop due to the placement of p3(n-1);

Directions: For each of the following problems, determine the output.

If there is an error or infinite recursion, then your answer should be "error"

(without the quotes).

5. Determine the output:

public class Main

{

public static void main(String[] args)

{

p1(3);

}

public static void p1(int n)

{

if (n < 0) {return;}

System.out.println(n);

p1(n - 1);

System.out.println(n);

p1(n - 2);

}

}

Output: 3  
MM§M2  
MM§M1  
MM§M0  
MM§M0  
MM§M1  
MM§M2  
MM§M0  
MM§M0  
MM§M3  
MM§M1  
MM§M0  
MM§M0  
MM§M1

6. Determine the output:

public class Main

{

public static void main(String[] args)

{

p2(3);

}

public static void p2(int n)

{

p2(n - 1);

if (n < 0) {return;}

System.out.println(n);

}

}

**Output**: error

7. Determine the output:

public class Main

{

public static void main(String[] args)

{

System.out.println(recursiveMethod(2, 3));

}

public static int recursiveMethod(int m, int n)

{

if (m == 0 || n == 0) {return 1;}

return Math.max(recursiveMethod(m, n - 1), recursiveMethod(n, m - 1)) + 1;

}

}

**Output**: 5