**Recursion Problems**

Directions - Calculate the output of these recursive problems. Your answers go in the box.

public int result(int n)

{

if(n==1) return 2;

else return 2 \* result(n-1);

}

What value does result(5) return?

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 mystery (3, 2, 6)?

public int f(int k, int n)

{

if(n==k) return k;

else if (n > k) return f(k, n - k)

else return f(k - n, n)

}

What value is returned by f(6, 8)?

public class Go {

public static int num = 2;

public static void main(String[] args)

 {

 go(42);

 System.out.println(num);

 }

public static int go(int n)

 {

 num +=n;

 if(n < 4) return 0;

 else if (n % 3 ==1) return go(n/3+2);

 else return go(n/2+1);

 }

What is the output of this program?

private static int product(int n) {

 if (n<=1) return 1;

 else if (n % 3 ==0) return n \* product (n-1);

 else return n \* product(n-2);

 }

What value is returned from product(6)?

public static void printIt(String s)

{

 int n = s.length();

 if (n < 1) return;

 String s1 = s.substring(1, n);

 printIt(s1);

 System.out.println(s);

 printIt(s1);

}

What would be the output of printIt("ABCD")?

public static void mystery(int a, int b)

 {

 System.out.print(a + " ");

 if(a <= b) mystery(a + 5, b - 1);

 }

What outputted by the call mystery(3, 21)?

Write a recursive method called takeSip that will take sips from a cup of coffee until it is empty. The method receives an integer value which represents the number of sips left in the cup of coffee. Here's the output for takeSip(4);

You take a sip. 4 remaining.

You take a sip. 3 remaining.

You take a sip. 2 remaining.

You take a sip. 1 remaining.

You take a sip. 0 remaining.

Coffee is all gone.

Here's your method header. You write the rest of the recursive method.

public static void takeSip(int n)

{

}