Recursive Methods

Objective: Trace recursive methods by hand.

Directions: Use a stack diagram to determine the effect of each of the following recursive methods. You will **receive credit only if you create the stack diagrams** to show your work. Do not just type the methods into your computer. That would NOT help you develop skill at solving these.

(A) Method:

```
public static int Factorial(int n)
{
        if(n==0)
        {
                return 1;
        }
        return n * Factorial(n-1);
}
```

Method Call:

```
System.out.println("Factorial");
System.out.println(Factorial(4));
```

(B) Method:

```
public static int Fibonacci(int n)
{
        if(n<2)
                return 1;
        return Fibonacci(n-1) + Fibonacci(n-2);
}
```

Method Call:

```
System.out.println("Fibonacci");
System.out.println(Fibonacci(4));
```

(1) Method:

```
public static void m1(int counter)
{
        if(counter==0)
                return;
        System.out.println("" + counter);
        m1(counter-1);
        return;
}
```

Method Call:

```
System.out.println("Problem m1");
m1(5);
```

(2) Method:

```
public static void m2(int counter)
{
        if(counter==0)
                return;
        System.out.println("hello" + counter);
        m2(counter-1);
        System.out.println("" + counter);
        return;
}
```

Method Call:

```
System.out.println("Problem m2");
m2(5);
```

(3) Method:

```
public static void m3(int x)
{
        System.out.print(x % 10);
        if(x/10!=0)
                m3(x/10);
        System.out.print(x%10);
}
```

Method Call:

```
System.out.println("Problem m3");
m3(100);
```

(4) Method:
```
public static int m4(int n)
{
        if(n==0)
                return 1;
        return 3*m4(n-1);
}
```
Method Call:
```
System.out.println("Problem m4");
System.out.println(m4(5));
```

(5) Method:
```
public static void m5(int a, int b)
{
        System.out.print(a+ " ");
        if(a<=b)
                m5(a+5, b-1);
}
```
Method Call:
```
System.out.println("Problem m5");
m5(10,5);
```

(6) Method:
```
public static void m6(int i)
{
        if(i<=8)
        {
                i=i+2;
                System.out.print(i + " ");
                m6(i);
        }
        else
                System.out.println("Who do we appreciate!");
}
```
Method Call:
```
System.out.println("Problem m6");
m6(0);
```

(7) Method:
```
public static void m7(String s)
{
        if(s.length()<8)
                m7(s+s);
        System.out.println(s);
}
```
Method Call:
```
System.out.println("Problem m7");
m7("star");
```

(8) Method:
```
public static int m8(int num)
{
        if(num<1)
                return 10;
        return num+m8(num-2);
}
```
Method Call:
```
System.out.println("Problem m8");
System.out.println(m8(20));
```

(9) Method:
```
public static int m9(int num)
{
        if(num>=20)
                return -5;
        return m9(num+4) + 2 * num;
}
```
Method Call:
```
System.out.println("Problem m9");
System.out.println(m9(20));
```

(10) Method:
```
public static int m10(int num)
{
	if(num>20)
		return -1;
	return num * m10(-2*num);
}
```
Method Call:
```
System.out.println("Problem m10");
System.out.println(m10(20));
```

(11) Method:
```
public static int m11(int x, int y)
{
	if(y==0)
		return x;
	return m11(y, x%y);
}
```
Method Call:
```
System.out.println("Problem m11");
System.out.println(m11(10,20));
```

(12) Method:
```
public static void m12(String z, int i)
{
	if(i<z.length()){
		m12(z, i+1);
		System.out.println(z.charAt(i));
	}
}
```
Method Call:
```
System.out.println("Problem m12");
m12("apple", 4);
```

(13) Method:
```
public static void m13(int num)
{
	if(num>0)
	{
		System.out.println("the message");
		m13(num-1);
	}
}
```
Method Call:
```
System.out.println("Problem m13");
m13(5);
```

(14) Method:
```
public static boolean m14(String s)
{
	if(s.length()>=3)
		return(s.charAt(0)==s.charAt(1) && s.charAt(1)==s.charAt(2) || m14(s.substring(1)));
	return false;
}
```
Method Call:
```
System.out.println("Problem m14");
if(m14("alphabet"))
	System.out.println(true);
else
	System.out.println(false);
```

(15) Method:
```
public static void m15(int p, int q)
{
	if(p/q==0)
		System.out.println(p+q+1);
	else
	{
		System.out.println(p);
		m15(p/q,q);
	}
}
```
Method Call:
```
System.out.println("Problem m15");
m15(10,20);
```

(16) Method:
```
public static void m16(int n)
{
	if(n==1)
		System.out.print("x");
	else if(n%2==0)
	{
		System.out.print("{");
		m16(n-1);
		System.out.println("}");
	}
	else
	{
		System.out.print("<");
		m16(n-1);
		System.out.print(">");
	}
}
```
Method Call:
```
System.out.println("Problem m16");
m16(5);
```

(17) Method:
```
public static int m17(int n)
{
	if(n==2)
		return 100;
	else if(n==3)
		return 200;
	else
		return (2*m17(n-1) + m17(n-2) + 1);
}
```
Method Call:
```
System.out.println("Problem m17");
System.out.println(m17(10));
```

(18) Method:
```
public static void m18(int n)
{
	if(n<=3)
		System.out.println(n+1);
	else
	{
		m18(n-3);
		System.out.println(">>" + (n-1));
	}
}
```
Method Call:
```
System.out.println("Problem m18");
m18(5);
```

(19) Method:
```
public static void m19(int n)
{
        if(n<=0)
                System.out.println(n);
        else
        {
                m19(n/3);
                System.out.print("," + n);
        }
}
```
Method Call:
```
System.out.println("Problem m19");
m19(5);
```

(20) Method:
```
public static int m20(int n)
{
        if(n>10)
                return n-2;
        n=n*3;
        return n+m20(n+2);
}
```
Method Call:
```
System.out.println("Problem m20");
System.out.println(m20(5));
```

(21) Method:
```
public static int m21(int n)
{
        if(n>5)
                return n-2;
        return n+m21(n+1);
}
```
Method Call:
```
System.out.println("Problem m21");
System.out.println(m21(5));
```

(22) Method:
```
public static int m22(int n)
{
        if(n<=1)
                return n;
        return n*m22(n-2);
}
```
Method Call:
```
System.out.println("Problem m22");
System.out.println(m22(5));
```

(23) Method:
```
public static int m23(int n)
{
        if(n<=0)
                return 10;
        return n+m23(n-1);
}
```
Method Call:
```
System.out.println("Problem m23");
System.out.println(m23(5));
```

(24) Method:
```
public static int m24(int num)
{
        if(num<1)
                return 10;
        return num+m24(num-2);
}
```
Method Call:
```
System.out.println("Problem m24");
System.out.println(m24(5));
```

(25) Method:

```
public static int m25(int num)
{
        if(num>=20)
                return -5;
        return m25(num+4) + 2 * num;
}
```

Method Call:

```
System.out.println("Problem m25");
System.out.println(m25(5));
```

(26) Method:

```
public static int m26(int num)
{
        if(num>20)
                return -1;
        return num*m26(-2*num);
}
```

Method Call:

```
System.out.println("Problem m26");
System.out.println(m26(5));
```

(27) Method:

```
public static int m27(int n)
{
        if(n==0)
                return 5;
        else if(n==1)
                return 8;
        return m27(n-1)-m27(n-2);
}
```

Method Call:

```
System.out.println("Problem m27");
System.out.println(m27(5));
```