Homework 13

Recursion!

1. Consider this code:

```
public static int power(int base, int power){
     if (power == 0){
          return 1;
     }
     else{
          return base*power(base, power-1);
     }
}
```

Make a recursion tree for `power(5, 4)`. Write down every call to factorial that would be made when evaluating `power(5, 4)` and it's result. You can use this format:

power(5, 4): 625

2a. Now consider this code:

```
public static int fibbonacci (int n){
     if(n == 1){
          return 1;
     }
     if (n == 2){
          return 1;
     }
     else{
          return fibbonacci(n-1) + fibbonacci(n-2)
     }
}
```

Make a recursion tree for fibbonacci(5). In addition, write down every call to fibbonacci that would be made when evaluating fibbonacci(5) and it's result. You can use the same format as above.

2b. Editors note: This is an incredibly innefficient way to do fibbonocis. Look at your recursion tree and write down why (hint: this isn't a question about recursion so much as a question about algorithms).

3. Write a recursive function that calculates factorials (5 factorial = 5*4*3*2*1 and is denoted "5!"). It should have the following signature:

```
public static int factorial (int n)
```

4. Write a recursive function that calculates the sum of the squares of the integers less than or equal to a given value. That value should be able to be an integer or a double, but the numbers whose squares are summed should always be integers. You can look Your function should have the following signature:

```
public static int sqsum(double n)
```

5. The AP exam writers absolutely adore recursive functions called mystery. The goal is usually to figure out what they do. Here are some AP-style problems involving recursive functions called mystery.

5a.

```
public int mystery(int n){
     if ( n == 1 ){
          return 2;
     }
     else{
          return 2*mystery(n-1)  }
}
```

What does `mystery(5)` return?

A. 64
B. 32
C. 16
D. 8
E. 2

5b.

```
public int mystery (int n, int a, int b){
     if(n == 0){
          return k;
     }
     else{
          if( n > k ){
               return mystery(k, n-k);
          }
          else{
               return mystery(k-n, n);
          }
     }
}
```

What valeue is returned by the call `mystery(6, 8)`?

A. 8
B. 4
C. 3

D. 2
E. 1

5c.

```
public int mystery(int k)
{
     if( k <= 0 ){
          return 0;
     }
     else{
          return (<missing code>);
     }
}
```

Which of the following could be used to replace <missing code> so that the value of `mystery(5)` is 15?

A. `k + mystery(k-1)`
B. `k * mystery(k-1)`
C. `mystery(k-1)`
D. `mystery(k+1)`
E. `mystery(k-1)*mystery(k+1)`

Warning: not all functions in questions like this are actually called "mystery." Some of them are called "result" or "k" or "f1" or equally nondescriptive things.

6. Class casting

6a. Write a function classCastUp which creates an object of type String and tries to cast it to type Object (it should only be a few lines long), and then print it using System.out.println(obj.toString()). Write what happens in a comment. If there is a runtime error, write the text of the error, and a breif interpretation of what the error means and why it happened. If there is a compile time error, record the text and interpret it, and then comment out the function so the rest of your code will compile.

6b. Write a function classCastAcross that creates an object of type String and tries to cast it to type Integer, and then prints it as above. Follow the instructions for 6a. as to how to document what happens.

6c. Write a function classCastDown that creates an object of type Object and tries to cast it to type String, and then prints it. Same as above.

6d. Write a function classCastUpDown that creates an object of type String, casts it to type Object and then back down and then prints it. Is the result the same as for 6c? Different?

7. Untyped ArrayLists

Choose any program you've written so far for this class which uses ArrayLists and rewrite it with untyped ArrayLists, using class casts every time you use the get method. Make sure your modified program works the same way as the original.