1. Give em the ’Ol Switcheroo

For each function call in the main method, write out the x and y values of both foobar and baz after executing that line. (Spring ’15, MT1)

```java
public class Foo {
    public int x, y;

    public Foo (int x, int y) {
        this.x = x;
        this.y = y;
    }

    public static void switcheroo (Foo a, Foo b) {
        Foo temp = a;
        a = b;
        b = temp;
    }

    public static void fliperoo (Foo a, Foo b) {
        Foo temp = new Foo(a.x, a.y);
        a.x = b.x;
        a.y = b.y;
        b.x = temp.x;
        b.y = temp.y;
    }

    public static void swaperoo (Foo a, Foo b) {
        Foo temp = a;
        a.x = b.x;
        a.y = b.y;
        b.x = temp.x;
        b.y = temp.y;
    }

    public static void main (String[] args) {
        Foo foobar = new Foo(10, 20);
        Foo baz = new Foo(30, 40);
        switcheroo(foobar, baz);  // foobar.x: ___ foobar.y: ___ baz.x: ___ baz.y: ___
        fliperoo(foobar, baz);   // foobar.x: ___ foobar.y: ___ baz.x: ___ baz.y: ___
        swaperoo(foobar, baz);  // foobar.x: ___ foobar.y: ___ baz.x: ___ baz.y: ___
    }
}
```
2 IntList to Array

For this problem we will implement a version of arraycopy that copies elements from an IntList into an array of ints. As a reminder, here is the arraycopy method:

```java
System.arraycopy(Object src, int sourcePos, Object dest, int destPos, int len)
```

System.arraycopy copies len elements from array src (starting at index source) to array destArr (starting from index dest).

To simplify things, let’s restrict ourselves to using only int[], and assume that srcList and destArr are not null. Additionally, assume that sourcePos, destPos, and len will not cause an IndexOutOfBoundsException to be thrown.

For example, let IntList L be (1 -> 2 -> 3 -> 4 -> 5) and int[] arr be an empty array of length 3. Calling arrayCopyFromIntList(L, 1, arr, 0, 3) will result in arr={2, 3, 4}.

```java
/** Works just like System.arraycopy, except srcList is of type IntList. */
public static void arrayCopyFromIntList(IntList srcList, int sourcePos, int[] destArr, int destPos, int len) {
    for ( ___________; ___________; ___________ ) {
        ___________________ = ______________________________;
    }
    for ( ___________; ___________; ___________ ) {
        ___________________ = ______________________________;
    }
}
```
3 Static Books

Suppose we have the following Book and Library classes.

```java
class Book {
    public String title;
    public Library library;
    public static Book last = null;

    public Book(String name) {
        title = name;
        last = this;
        library = null;
    }

    public static String lastBookTitle() {
        return last.title;
    }

    public String getTitle() {
        return title;
    }
}

class Library {
    public Book[] books;
    public int index;
    public static int totalBooks = 0;

    public Library(int size) {
        books = new Book[size];
        index = 0;
    }

    public void addBook(Book book) {
        books[index] = book;
        index++;
        totalBooks++;
        book.library = this;
    }
}
```

(a) For each modification below, determine whether the code of the Library and Book classes will compile or error if we only made that modification, i.e. treat each modification independently.

1. Change the totalBooks variable to non static
2. Change the lastBookTitle method to non static
3. Change the addBook method to static
4. Change the last variable to non static
5. Change the library variable to static
(b) Using the Book and Library classes from before, write the output of the main method below. If a line errors, put the precise reason it errors and continue execution.

```java
public class Main {
    public static void main(String[] args) {
        System.out.println(Library.totalBooks); // Error reason
        System.out.println(Book.lastBookTitle()); // Error reason
        System.out.println(Book.getTitle()); // Error reason

        Book goneGirl = new Book("Gone Girl");
        Book fightClub = new Book("Fight Club");

        System.out.println(goneGirl.title); // Error reason
        System.out.println(Book.lastBookTitle()); // Error reason
        System.out.println(fightClub.lastBookTitle()); // Error reason
        System.out.println(goneGirl.last.title); // Error reason

        Library libraryA = new Library(1);
        Library libraryB = new Library(2);
        libraryA.addBook(goneGirl);

        System.out.println(libraryA.index); // Error reason
        System.out.println(libraryA.totalBooks); // Error reason
        libraryA.totalBooks = 0;
        libraryB.addBook(fightClub);
        libraryB.addBook(goneGirl);

        System.out.println(libraryB.index); // Error reason
        System.out.println(Library.totalBooks); // Error reason
        System.out.println(goneGirl.library.books[0].title); // Error reason
    }
}
```