1. (3 points) What causes null pointer exceptions?

2. (3 points) Why is it important to keep your code simple?

3. (4 points) Why can drawing a picture help you solve a problem you are stuck on?

4. (4 points) What does foo mean when used as an identifier?

5. (6 points) Write a loop that will print the numbers from 1-10, one per line.

6. (30 points) Write a complete Account class with a single int variable, balance, along with its accessors, toString() and an initializing constructor.
7. (10 points) Write an `AccountList` class (that extends `ArrayList<Account>`) with a `public String toString()` method which returns a header (like "AccountList:") and all its Accounts, one per line (assume `Account` has a `toString()` method, as in previous problem).

8. (10 points) Assume there are two globally defined `AccountLists`, big and little. Write a `split(AccountList, int)` method which will empty both big and little, and then add all the Accounts with balances bigger than the int parameter to big and the rest to little.
9. (10 points) Finish the `Patron(MyReader)` constructor, below. Assume that the format of the file associated with the `MyReader` is the patron's name, followed by the titles of the books they have out, one per line, terminated by a blank line. Like this:

Alice
Moby Dick
War and Peace
Bob
The Cat in the Hat
Danni
Joe

As you can see, Alice has two books, Bob, one, whereas Danni and Joe have none. Recall that `MyReader` has two methods, `boolean hasMoreData()`, and `String giveMeTheNextLine()`. Assume that there is a `Book find(String)` method which will return the `Book` whose title which is passed as a parameter or null if it is not found -- if any titles are not found, `sout` an error message. Assume `BookList extends ArrayList<Book>`.

class Patron {
    String name;
    BookList checkedOutList = new BookList();

    Patron (MyReader mr) {
10. (10 points) Write a method, reverse, which returns its String parameter backwards. So if the parameter were "linaf", it would return "final".

11. (10 points extra credit) Assume you a a global BookList theBookList. Write a method Book find(String) that returns the Book in that list which has the title passed as the parameter, or null if there is no Book with that title. Assume standard accessors as needed.