Students:

This content is controlled by your instructor, and is not zyBooks content. Direct questions or concerns about this content to your instructor. If you have any technical issues with the zyLab submission system, use the **Trouble with lab** button at the bottom of the lab.

20.26 Ritchie: Program 6: Create Your Own Program

Program 6 Create your own Program

Due: Wednesday 12/11/19 , 11:59PM

Program Design

In this assignment you will be designing and implementing your own program. This means you must come up with a program that accomplishes some task or goal. For example, perhaps your program is a book cataloger, or a student waitlist, etc…

At a minimum, your program must contain the following:

- 3 Classes
- 1 Abstract Class (This means a class must inherit from this abstract class)
- 1 Use of Generics in a class or abstract class
- 1 Interface (This means a class must implement this interface)
- 1 Use of Recursion (This means you must use recursion at least once in a class, or abstract class)
- Implement 1 Sorting Algorithm
  - You get to choose which sorting algorithm
  - Do not using an existing class to this, you must implement the algorithm from scratch
- 1 Use of Binary Search (This means you must use recursion at least once in a class, or abstract class)
  - Do not use an existing class to this, you must implement the algorithm from scratch
- 1 Use of an iterator
  - This can either be explicitly or implicitly (i.e. using an enhanced for loop will suffice)
  - You do not need to implement an iterator (Unless you want to) simply use one from an existing data structure
- 1 Use of one of these data structures: Queue, Stack, LinkedList
  - You do not need to implement the data structure (Unless you want to) simply use one from Java's Collections Interface

Readme File

In Software development it is common practice to include something called a: "Readme" file that describes what a piece of software does and how to use it. In your case, create this file: Readme.txt. In this file include the following:

- Your name, red Id, and the date
- A description about what your program does in general
- A brief summary about what each file in your program is doing

Unit Tests

For this program you will be writing your own unit tests. Usually in software engineering you would use Unit Testing framework, for example Java has JUnit5, but for this assignment you will simply use a main (more on this in a second). Writing unit tests can be tricky, but the idea is to think of edge cases that might break your code, and to use those edge case conditions as a unit test to see how your program handles extreme circumstances.

Note your UnitTests class does not count towards the number of classes I have required for this assignment.

At a minimum you must write:

- 2 tests for each class, abstract class, and interface
- 1 test for Binary Search
- 1 test for Recursion
- 1 test for your Sorting Algorithm

Note the above specifications mean testing the functionality, not implementing that functionality in the test itself. So for example, do not implement Binary Search inside of a test, instead Binary Search should be apart of a method in some class, and you test that method to see if it yields expected results.

If you do recursion in conjunction with your sorting algorithm or with binary search, then you can remove the unit test for recursion.

Since you are not using a Framework you will create your unit tests by:

- Create a class called: UnitTests
- In this class create a main method:
  - public static void main (String args[ ]) … etc
- In this class each "Unit Test" should be a static method that gets called in main. End each method name with the word: "Test".

For example this would be considered a single test:

```
public static void main(String args[ ]) {
    addFiveTest();
}

public static void addFiveTest( ) {
    Foo bar = new Foo(5);
```

```
    bar.addFive( );
    if (bar.getX() != 10) {
        System.out.println("FAILED ADD FIVE");
    }
    else {
        System.out.println("SUCCESS ADD FIVE");
    }
}
```

Program Turn in and Grading

Since zybooks will not let me set this up properly, you will turn in this program by emailing me a compressed zip file containing all of your program files. If you email me anything other than a compressed zip file I will not accept it. As a reminder my email is: writchie@sdsu.edu, you can also find this in the syllabus.

I will accept the latest email you send me as the program to grade. With that said I will be excepting no more than 5 submissions from you. In other words if you submit this assignment to me more than 5 times I will only grade the fifth submission and no submission after.

There will be no late turn in for this assignment….

Grading is as follows:

- 50% Following program design specifications
- 10% Readme file
- 30% Unit Tests
- 10% Good comments, and well formatted code

Your code MUST COMPILE!!! Do not turn in code that cannot compile. If you do that will result in an immediate ZERO for this assignment.

I may assign extra credit points if I feel a project is particularly well thought out an implemented.

LAB ACTIVITY 20.26.1: Ritchie: Program 6: Create Your Own Program 0 / 1

Submission Instructions

Upload your files below by dragging and dropping into the area or choosing a file on your hard drive.

Submit for grading

Latest submission

No submissions yet

Trouble with lab?