CSCI 2070 Assignment 5

Q1: **Triangle Class** [30 points]
Make a `Triangle` class that has three fields:

- One for the base of the triangle (in double): `base`
- One for height of the triangle (in double): `height`
- One for the name of the triangle (in String): `name`

The class should have:

- A method that returns the triangle's area: `(base*height)/2`
- An `equals` method: returns true when two triangle objects have the same area, otherwise false
- A `toString` method: prints the base and height of the triangle

Demonstrate the class in a program that asks the user to enter the name and the dimensions for two triangles. The program should:

- Display the area of each triangle
- Indicate whether the triangles are of equal size
- Print the triangles using the `toString` method

Note: For this question, name your java class `<my name>Triangle`. For example, my java class would be named: `ChenchuttaJacksonTriangle`. Take note of requirements for arguments and returns.
Hence, the java filename should be `ChenchuttaJacksonTriangle.java`.
You also need to write a separate program `<my name>TriangleDemo.java` to test the class.
Important: If you do not put `<my name>` to the above mentioned fields (class name and filename), **you will get 0 point for the question.**

Estimated time: 2 hours

Q2: **Solid** [30 points]
Design a `Solid` class with the following methods:

- A static method that accepts the radius of a sphere and returns the volume of the sphere.
  - Use the following formula: *volume* = (4/3) * π * r^3
  - Use `Math.PI` for π and the radius for *r*
- A static method that accepts the length, width, and height of a right rectangular cuboid (rectangular prism) and returns the volume of the cuboid.
  - Use the following formula: *volume = Length * Width * Height*
- A static method that accepts the base area of a cone, and the cone's height. The method should return the volume of the cone.
  - Use the following formula: *volume* = (1/3) * *b* * *h*
  - Use base area for *b* and height for *h*.

The methods should display an error message if **negative values** are used for the sphere's radius, the right rectangular cuboid's length, width or height, or the cone's base area or height.

Next, write a program to test the class, which display the following menu and responds to the user's selection (ask for user input, and display output etc):

```
Volume Calculator:
1. Calculate the volume of a sphere
2. Calculate the volume of a rectangular prism
3. Calculate the volume of a cone
4. Quit
Enter your choice (1-4):
```

Display an error message (and show the menu again), if the user enters a number outside the range of 1 through 4 when selecting an item from the menu.

Note: For this question, name your java class `<my name>Solid`. For example, my java class would be named: `ChenchuttaJacksonSolid`.
Hence, the java filename should be `ChenchuttaJacksonSolid.java`.
You also need to write a separate program `<my name>SolidDemo.java` to test the class.
Important: If you do not put `<my name>` to the above mentioned fields (class name and filename), **you will get 0 point for the question.**

Estimated time: 3 hours

Q3: Textbook Page 698, Chapter 10, Programming Challenges 1, **Employee and ProductionWorker Classes** [40 points]
Your program needs to be able to check if an "Employee Number" is in the valid format (XXX-L). Demonstrate the checking in your test case.
Note: For this question, name your java classes `<my name>Employee`, and `<my name>ProductionWorker`. Hence, the java filenames should be `ChenchuttaJacksonEmployee.java`, and `ChenchuttaJacksonProductionWorker.java`.
You also need to write a separate program `<my name>WorkerDemo.java` to test the class.
Important: If you do not put `<my name>` to the above mentioned fields (class name and filename), **you will get 0 point for the question.**

Estimated time: 3 hours

Submission instructions:
You need to compile the above programs (questions) separately, and provide **two test cases** (if applicable) for each program (question). Do a screen capture of the input and related output for each test case. Use any graphic editing software (e.g. Microsoft Paint, Adobe Fireworks) to cut out the program input and output (from the screen capture), paste them into a word document under a related question number, save the document as a pdf file. A sample input/output (screen capture) can be found at the end of this document.

You need to submit the following:
1. A pdf file containing the screen captures of program input and output of all test cases, name the file **lastname_firstname_assignment05.pdf**.
2. All java files (.java files only). Zip your java files into a single zip file (or rar file) **lastname_firstname_assignment05.zip**.

Please submit electronic copy (the above mentioned **two files**) to D2L digital dropbox.

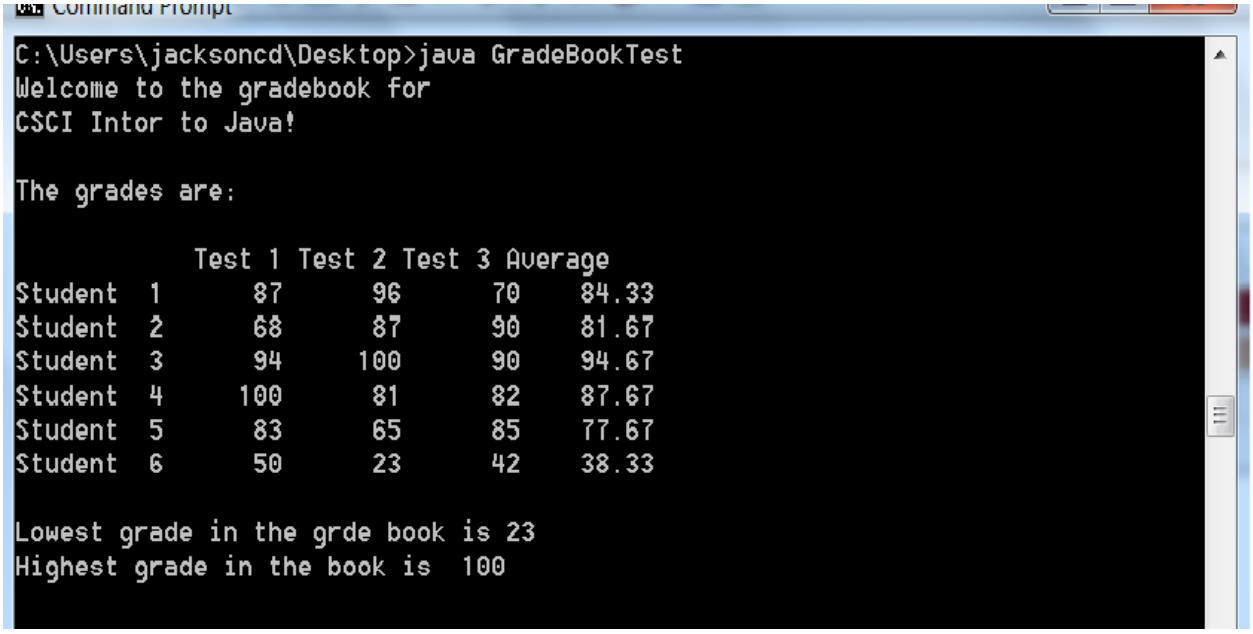
Grading guidelines (programming questions):
Your programs will be judged on several criteria, which are shown below.

- Correctness (50%): Does the program compile correctly? Does the program do what it's supposed to do?
- Design (20%): Are operations broken down in a reasonable way (e.g. classes and methods)?
- Style (10%): Is the program **indented** properly? Do variables have **meaningful names**?
- Robustness (10%): Does the program handle erroneous or unexpected input gracefully?
- Documentation (10%): Do all program files begin with a **comment** that identifies the author, the course code, and the program date? Are all the classes, methods and data fields clearly **documented (comments)?** Are unclear parts of code **documented (comments)?** (Some items mentioned may not apply to some languages)

A program that does not compile will get at most **50% of the possible points**.

Sample input/output (screen captures)

Question X, test case 1, input/output:



Screen capture must be readable by the instructor, or 0 point will be given for the question.
Please note that you can use more than one screen captures for each test case.