Comp 170 Project Proposal.

You will define, create, invent, program, test, and review a java program to do something you pick!

1. Teamwork: optional, you may be alone or teams of up to 4. Teams must do more work than individuals. If a team you will need to define how to divide the work.
2. Project Name: you need a name.
3. Progress Reports: may be required during work time on the project
4. What Will You Do?
   a. You can do anything
   b. Some examples will be provided
   c. Don't worry, I'll help you decide if it is too easy or too hard!
5. Technical Requirements. Your finished project must include *at least* these items
   a. Three classes that implement true objects. One of these must be a container (array, list, dictionary) to include other objects.
   b. Complete pseudo code
   c. Well structured code in all parts (classes, methods, main). Use function/subroutines/methods to divide up the code into understandable chunks. Avoid long, unstructured code in any single method.
   d. A user interface that allows full demonstration and testing of the app.
   e. Note: if some of these are really impossible for your project, discuss with me. I want you to be able to do something interesting, original, and enjoyable!
6. All code produced must use appropriate style as used in the class.
7. You will present and demonstrate your finished project to the class

What do you need to turn in? See the outline below and include all parts of that in your project proposal. You can include more parts and ideas if you wish.

V2.0 Java Edition November 2016.

Project Proposal Outline

Complete these items for your project plan:

Project Title (make it interesting)

My App Definition

A paragraph or two describing what the app does and why it's interesting. Some also call this your "elevator pitch" - if you happen to bump into someone in the elevator and you have 30 seconds before they get out, how do you sell them on your great idea for an app.

Team or Individual
If you plan a team, list all members and possible roles on such things as design, programming, team coordination, documentation, testing, final report creation.

Design

What are the key components you plan to use? What kinds of input and output will you need from the user or from files/

What will the user interface look like? You can mock it up in text to give an example use case.

Initial ideas for classes. These are often the nouns in the App Definition. A couple of ideas of things the classes should be able to do (these may be the verbs in the App Definition).

Anything you are not yet sure how to do?

Schedule

Make a 4 week plan to do the work. List key activities for each week:

Week 1 (Monday start date) Key things to do

Week 2 (Monday start date) Key things to do

Week 3 (Monday start date) Key things to do

Week 4 (Monday start date) Key things to do

Testing

How will you test your app? Do you plan to have anyone other than you try it (you should!)? If you want to try out some software engineering concepts, keep a list of each of the bugs your find while testing and how long you spent testing. Make that part of your project demonstration.