Project Goals

- Summarize all of your course knowledge in one significant coding project.
- Work in a team to implement an interesting system.
- Research a new data structure that we won’t cover during this semester.

Assignment Details

A significant part of your overall course grade (10%) is linked to the course project. There are three options that you can select from to complete your project. Each is detailed in the following sections.

For your project, you will work in groups of two or three (preferably), but each member of the group will be evaluated separately based on his or her contributions to the project. This evaluation will be determined largely from the feedback of the other group members. You may also choose to work on your own if your schedule will not allow you to meet with other group members regularly. However, I would discourage this option.

Project Option 1: Card Game

In this project option, you will implement some card game such that a user can play against the computer. The sole requirement of this project is to implement a Card class, so that each Card has a value property (\{2, 3, 4, 5, ..., ‘J’, ‘Q’, ‘K’, ‘A’\}) and a suit property (\{“CLUB”, “HEART”, “DIAMOND”, “SPADE”\}). You will need to create a deck of Cards stored in a data structure (a Queue or a Stack would be the most natural options), so that the Cards can be shuffled, dealt in a specified order, and that each Card only exists once. Aside from this requirement, the game and how you choose to implement it is up to you.

The two easiest card games to implement are poker and blackjack, though you may certainly choose to implement another, so long as a user is competing against the computer. You will need to research the rules of each of these games to ensure that your implementation is correct.

Using the poker game as an example, you will create your Card data structure, and write a function to deal 5 Cards to the player and the computer. The user then will have to option to keep or discard any of his/her Cards, so you will need to write a function to handle this. Similarly, the computer will have the option to keep or discard any set of Cards. You can make this AI as...
simple or as detailed as you would like. Finally, you will need a function to determine what kind of hand the user and the computer have (royal flush, flush, straight, etc.), noting that precedence is important because a user with a full house also has a set of 3 and a set of 2. You will then need a function to decide if the user or the computer has won, based off of the same precedence. If both the user and the computer have matching precedence levels (both have a full house), then you’ll need to write a tiebreaker function to determine that a queens and 8s full house beats a jacks and 9s full house.

Though you do not need to submit a proposal for this project option (see Timeline and Deliverables section), you should still notify me of what card game you plan to implement.

**Project Option 2: Research Paper**

If you are tired of writing code and want to do something different, you have the option of researching a data structure or algorithm that we will not have time to discuss during this semester. Some proposed topics are listed at the end of this section, or you may propose your own topic. Students in CMPSC250 who choose this option should *not* select an algorithm or data structure we discussed in that course – you must do new research.

If you select this project option, your writing deliverables will be more significant and graded more harshly that the other project options. You can use the textbook as a primary resource for your research, but you must also include at least 4 other resources, 2 of which must *not* be Web resources.

Please note that groups for this project choice are limited to 2 members – no 3 person research groups will be permitted.

Though you do not need to submit a proposal for this project option (see Timeline and Deliverables section), you should still notify me of what data structure or algorithm you plan to research.

**Proposed topics:** Skip List, Red-Black Tree, Boyer-Moore Algorithm, Knuth-Morris-Pratt Algorithm, Splay Tree, Bellman-Ford Algorithm

**Project Option 3: Propose Your Own**

If you choose to implement your own proposed project idea, you should select something with real-life applications or that tests some series of widely-used algorithms or data structures. You have a significant amount of flexibility with what you can do with your project, so use it wisely! Is there something you wanted to create but never had time? Is there a project from other courses that you always wanted to extend? Juniors: is there an idea you have for your thesis that you want to try out first before committing to it? Seniors: can you supplement your thesis with a programming aspect?

There are a limited number of hard requirements that you must meet if you select this project option. Aside from meeting these, you are free to take your project wherever you would like.
1. Your project must be approved by me before you start working on it.

2. Your project must be written in Java.

3. Your project must solve a real-life problem or develop a real-life application. You need to research the problem you select to get an idea of what has already been done. You must include references to existing work in your final report.

4. You must provide some mathematical analysis (Big-O notation) of the running time of some component of your system. Ideally, this should be some major or centric component to your system (i.e. not an analysis of sorting a list when you are reading in data). We should discuss which component of your system this will be at around the time you submit your Progress Report.

5. Your project must have a significant implementation part where you will develop program(s) for your chosen problem. You may write your code from scratch, or reuse and extend some existing code. Obviously, anything you use that is not yours must be documented.

6. Your project must be extensive enough to qualify as a project (think of work for at least 4 one-week lab assignments), but not too extensive so that you cannot finish it in the remainder of the semester (one month).

Timeline and Deliverables

1. Group and Project Selection, Deadline April 1
   By Friday, April 1, you should have identified your group and project that you will be working on, and notify me by email so that I have record of your selection. If you choose the third option to propose your own project, you should also submit a 1-2 page project proposal at this stage for feedback.

2. Progress Report – 2-3 pages, Deadline April 15
   By Friday, April 15, you should submit a brief progress report detailing the work that you have completed on your project so far. Have you experienced any unexpected challenges? Did you have to change anything major in your implementation as you were working? Include everything you have done so far in your progress report, even if it is incomplete. No need to include the actual code (unless you want my help with it), just describe what progress you have made with it.

3. Presentation, Deadline April 27
   By the presentation session during our final lab session, you should have finished your implementation or your research. In the presentation, you should describe at a high-level what you have achieved with your project during the past month. The goal of the presentation is to convey the important ideas rather than be a formal specification of everything you did. Prepare for a ~10 minute presentation. Design at least 5 slides, including a slide with the title of your project and group members.
   Every member of the group needs to contribute to the presentation talk. Also, you need to show a demo at the end of the presentation of your system (if applicable).

Incorporate any feedback from the progress report and the presentation session (if applicable). Your final report should be clear and well written, which includes no typos or grammatical errors. The writing will be graded more harshly here. Your report should be written in a professional and technical manner.

Submission Details

For each deliverable, you need to submit a PDF with your report (or presentation slides). For your final report, you need to submit any supplementary material (code, data, a README file documenting what everything is and how to run your program) to a BitBucket repository dedicated to your project.

1. Group and Project Selection (and Proposal if applicable) – 10 points
2. Progress Report – 25 points
3. Presentation – 30 points
4. Final report (and Implementation if applicable) – 35 points

Please remember that all files that you submit should be your own work, though you are welcome to discuss high-level topics and algorithms with classmates.