Semester (2011 Fall) Challenge/Bonus Projects

(**Due on Tuesday, 12/13/2011 at Moodle**)

(**Presentation in class on Friday, 12/9/2011—***not required but encouraged*)

Your name:	Score:

Important Notice:

1. **You must submit your work at Moodle**.
2. How to submit:
    - Create a folder for the assignment;
    - Create **a subfolder for each** of the projects;
        i. This subfolder ***MUST start with "ChallengeProjNN"—NN will be 1, 2, …13, denoting the project numbers*** (see Project Lists)..
    - Copy the projects to their corresponding subfolders;
    - Compress the folder and its contents into a SINGLE file;
    - Upload the compressed file at Moodle.

For the following bonus/challenging projects, you may need to use the topics we have not covered (will or will not cover) in the class. But at present, you have enough knowledge/preparation to make use of the sample code related with a certain topic from the textbook, online source, and particularly Java online Docs/Specs on how to use classes!!—Remember, the quickest way (and the best) to *learn is starting from the examples*! The following gives a basic quick reference (the textbook **is Java How to Program, 8th edition—***if you don't have this version, I'll make copies of the question description*):

- Chapter 7: Arrays (go through all its examples—required!!)
- Chapter 18: Recursion (go through all its examples—required to sharpen your problem-solving skills!!)
- Chapters 12 & 13: OOP Case Study (very good example on designing/implementing a relatively complicated system—recommended)
- Chapter 16: Strings/Chars/Regex (recommended)
- Chapter 17: Files (optional—if your project deals with file I/Os, then required)
- Chapter 14: GUI Part I (required—since most of the projects do need GUI display)
- Chapters 25 & 15 (GUI part II & Java 2D—optional but strongly recommended)
- Chapter 23: Applets/Java Web (strongly suggested—for posting your project/demo to the website—you may just need *a couple of minutes* to cover this chapter!!)

Project Lists (you must talk to me *before starting on the projects*—**credits** of each project vary from 1/challenge point (and 50 assignment points) *to infinite*….and can be *negotiated*!!)

1. Application of Monte Carlo method in calculating π value (animated GUI required).
2. Simulation of a relatively complicated calculator.
3. GUI-Based Craps Game (check question 14.16 of textbook, on p. 636)
4. GUI-Based Shape Project
    - First complete GUI Case Study Exercises 10.1 and 10.2 (on p.436~438 of the textbook).
    - Then Complete Exercise 14.17 on p. 637~638.
5. Java2D Project I—Shapes and Random Colors: complete exercise 15.29 of textbook on p.679
6. Java2D Project II—Select Shapes and Color Dialog: complete exercise 15.30 of textbook on p.679
7. GUI & Java2D Project I—Java2D Drawing: complete exercise 15.31 of textbook on p.679~680
8. GUI & Java2D Project II—Large -type Display: complete exercise 15.32 of textbook on p.680
9. Turtle Graphics:
    - First complete exercise 7.20 of the textbook on p.303
    - Then complete exercise 15.23 of the textbook on p.678
10. Tortoise and Hare:
    - First complete exercise 7.28 of the textbook on p.307~308
    - Then complete exercise 15.25 of the textbook on p.678
11. Knight Tour:
    - First complete exercises 7.22, 7. 23 and 7.26 of the textbook on p.304~307
    - Then complete exercise 15.24 of the textbook on p.678
    - (You may figure out different heuristics for this project—**can be very challenging!!**)
12. 8-Queens (or, start from 8-Queens and then work on n-Queens, such as n=16,.. 100000…):
    - First complete exercises 7.24 and 7.25 of the textbook on p.306~307
    - Then design/implement a GUI for this project
    - (You may figure out different heuristics for this project—**can be very challenging!!**)
13. Any other Java GUI/Java2D projects such as
    - Sudoku (or a simplified Sudoku)
    - 8-puzzle (and then 16 puzzle…)
    - Connect 4
    - Design a logo for SSU's Programming Club
    - ….(any others, such as Fractals…)….