CS 351 Design of Large Programs

Spring 2017

Instructor:

Joel Castellanos

Office: Electrical and Computer Engineering (ECE) Room 233
Office Hours: Tue/Thurs 3:30 PM – 5:30 PM and by appointment.
e-mail: joel@unm.edu
GitHub: castellanos70

Lab Instructor:

Course Web site: http://cs.unm.edu/~joel/cs351

Textbook:

Selected readings posted on course website or electronic reserve.

Description:

This class is about designing big software, where big refers to projects with a scope too large to be handled by any one person at any one time. This course primarily deals with software design, time management, and strategies for completing complex coding tasks.

Programming Language and Environment:

All programming examples and assignments will use Java 1.8 JDK, the IntelliJ IDE (the free Community version is fine), Git source control and GitHub repository (there are free student accounts that can be made private). As most of the work in this course will be done in group projects and as the groups will switch from project to project. It is important that everyone is used to the same coding standards IDE, and source control.

Grading:

- 60%: Individual and Group Programming Projects.
- 20%: Class Participation (quizzes, lab period code reviews, project presentations and discussions).
- 10%: Midterm exam (short essays on reading).
- 10%: Final exam (short essays on reading).

Working Together:

Of course, in a group project, you will share code with your group. Whatever part of the code to which you claim to have made a significant contribution, you must be able to explain. On individual assignments, you are NOT permitted to share code or electronic copies of code. You are, however, encouraged to help each other by discussing *specifications*, *algorithms*, *data structures*, and *test cases*.

Turning in Lab and Project Assignments:

If you want to receive credit for your hard work:

1) Each programming assignment must be attached in Blackboard Learn. All group project turn-ins must attach a Git repo link. Individual projects may attach either a .JAR or a repo link. E-mailed programming assignments will be reviewed for feedback, but then deleted without being graded or recorded.

2) Each programming assignment must be named in the form:
   *s*_*name1*_*name2*_[*name3*_[...]].jar

   Where, *s* is a descriptive string, and the underscores are literal. For individual projects, *name1* and *name2* are your first and last name. For group projects, each group member's **first** name must be included and delimited by the underscore character.

3) The .jar file turned in with each programming assignment must include all required source code (.java files) within their required package structure. NOTE: Including source is ***not*** the default option.

4) The .jar file either attached or present in at the top level of the linked repository must be an ***executable .JAR*** that ***is able to find all needed resources*** (other than the JDK 1.8) ***within that .JAR***.

5) Each assignment with more than one source file must include a README.txt file that explains: how to use the program (which class contains the entry point, command line arguments, ...) , who did what parts of the program and what parts (if any) are by third party.

6) ***Each member*** of a group must attach in Blackboard Learn a link to your group repository (this is needed so feedback in Blackboard can be given to student individually).

7) Each program must compile using Oracle's Java SE 1.8 SDK.

8) Your programs will be graded using the CS lab computers running Ubuntu in CENT B146. Well written Java code should be platform independent. That said, it would be wise to always test your .JAR on one of the designated CS lab computers.

Attendance:

Attendance is required for both lab and lecture. During lab and during lecture, attendance is taken in the form quizzes, code reviews, and discussions.

Title IX:

In an effort to meet obligations under Title IX, UNM faculty, Teaching Assistants, and Graduate Assistants are considered "responsible employees" by the Department of Education (see pg 15 - http://www2.ed.gov/about/offices/list/ocr/docs/qa-201404-title-ix.pdf). This designation requires that any report of gender discrimination which includes sexual harassment, sexual misconduct and sexual violence made to a faculty member, TA, or GA must be reported to the Title IX Coordinator at the Office of Equal Opportunity (oeo.unm.edu). For more information on the campus policy regarding sexual misconduct, see: https://policy.unm.edu/university-policies/2000/2740.html

ADA:

In accordance with University Policy 2310 and the Americans with Disabilities Act (ADA), academic accommodations may be made for any student who notifies the instructor of the need for an accommodation. If you have a disability, either permanent or temporary, contact Accessibility Resource Center at 277-3506 for additional information.

Syllabus Weekly Schedule

Week 1:	Creating an executable JAR with imbedded resources. A* Pathfinding.
Week 2:	JavaFX Polymorphism, Interfaces, Inner Classes and Collections Multithreaded Programming in Java
Week 3:	Design Patterns (I) Version Control with Git Conway's Game of Life on a massive grid in parallel
Week 4:	Start of Group Project I: Zombie House Design Patterns (II)
Week 5:	Design Patterns (III)
Week 6:	Design Patterns (IV) and UML Diagrams
Week 7:	Unit Testing
Week 8:	Midterm Exam
Week 9:	Start Group Project: Starvation Evasion Selected readings on Object Oriented Programing Design
Week 10:	Socket Communication and Client Server Models
Week 11:	Selected readings on efficient programming in Java (I)
Week 12:	Selected readings on efficient programming in Java (II)
Week 13:	Designing Artificial Intelligence
Week 14:	Hill Climbing and Adaptive Hill Climbing (local search)
Week 15:	Genetic Algorithms (global search)
Week 16:	Final Group Project Presentations