CS 3354.253: Object-Oriented Design and Programming

Spring 2019

Instructor: Dr. Jelena Tešić (pronounced as Yeh-LE-nah TE-shich)
Office Number: CMAL 307B
Telephone (Office): 512.245.3436
Email Address: jtesic@txstate.edu
Faculty Profile: https://cs.txstate.edu/accounts/profiles/j_t463/
Office Hours: Mon Wed 3:30 – 5 p.m. and by appt

Section Information: CS3354-253 IGRM 03104
Class Meetings: Mon Wed 2:00 p.m. - 3:20 p.m.
Open Labs: DERR 231 (Linux Lab) MCS 590 (Windows Lab)
Prerequisites: Grade of C or better in CS 2308

Course Material

1. Class slides, notes, and source code posted on TRACS, No required textbook
2. Suggested Textbooks:
   - HEAD FIRST SERIES: JAVA, OBJECT-ORIENTED ANALYSIS, and DESIGN PATTERNS
   - Please contact the instructor for more recommendations.

Course Description

The course covers object-oriented design principles and programming for students with prior programming experience. The topics include inheritance and polymorphism, object-oriented design process, UML diagrams, design patterns, exception handling and multithreading. Students will design and implement programs in Java.

Course Objectives

At the end of the semester the student should be able to:
- Design, implement, test, and debug programs in an object-oriented programming language: Java.
- Describe the unique features of Java.
- Read and write Java programs that use generic types and data types from the Java Collections library.
- Describe the concepts of inheritance and polymorphism and incorporate them into Java programs using abstract classes and interfaces.
- Describe how the class mechanism supports encapsulation, information hiding, and interfaces.
- Develop programs using multiple classes and composition.
- Read and write Java programs that use persistence (serializable objects).
- Describe and apply the Object-oriented design process to design a Java program.
- Describe the semantics of exception handling in Java, and use it to write reliable Java code.
- Read, design, and draw the following models using the Unified Modeling Language (UML)
- Write Java code that implements the designs specified by UML diagrams.
- Describe the following Design Patterns and create UML designs using them and implement the designs in Java programs.
- Determine the proper design pattern for a given problem.
- Use Javadoc to specify the interface (API) of Java objects.
- Understand and apply event-driven programming principles by developing programs with a graphical user interface, using objects from the Java Swing library.
- Read and write Java programs that use threads to implement concurrency.
What is expected of student registered for CS 3354.253 in Spring 2019?

Students are expected to:
1. Attend instructional meetings
2. Do not distract or disrupt students during instructional meetings
3. **Read announcements** from the instructor posted on TRACs course site.
4. Be informed and prepared for the class
5. **Submit homework assignments on time**
6. Complete individual assignments by yourself. Help will be provided during office hours of an instructor and a teaching assistant.
7. Participate in the completion of a group assignment. Group assignment means the entire group gets an identical grade score.
8. Check assignment grading, assessment grading, and grade grievance policy
9. Take midterm and final exam in the classroom during the exam time
10. Clearly communicate with the instructor regarding issues, delays or unforeseen circumstances in timely manner. Emailing is the fastest way to reach the instructor.

### Grading

<table>
<thead>
<tr>
<th>What</th>
<th>Grade Percentage</th>
<th>Date (if applicable)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Homework and Programming Assignments</td>
<td>50%</td>
<td>Due date in TRACs and announced in the class</td>
</tr>
<tr>
<td>Midterm</td>
<td>15%</td>
<td>Mon Mar 11th 2 p.m. – 3:20 p.m.</td>
</tr>
<tr>
<td>Final</td>
<td>30%</td>
<td>Monday May 13th 2 p.m. - 4:30 p.m.</td>
</tr>
<tr>
<td>Quizzes and Class Participation</td>
<td>5%</td>
<td>Class participation and quizzes through TRACs Assessment Tab</td>
</tr>
</tbody>
</table>

### Assignments

- Assignments will be announced on TRACs.
- All assignments must be done either individually or in pairs or in groups, instructor will clearly specify the grouping for each assignment.
- The homework assignments involve drawing models and providing some written explanations. The programming assignments involve developing programs in Java. You will submit your assignments using TRACs or github, as specified in the Assignment announcement.
- Late assignments will incur 10-point (out of 100 point) penalty per day for up to 3 days
  - Students get 5 freebies – 5 late assignment submission dates with no penalty
- After 5 days, no submission will be accepted. Last day of will be posted on TRACs.
- Submission time is always 11:55 p.m. on a given date (Friday) and submission closes on 11:5 p.m. 5 days later (Monday).

### Policies

**Grade Grievance Policy:** If a student believes a mistake has been made in grading an assignment, the student has one week after an assignment is returned to resubmit an assignment for re-grading if they believe there is an error.

**Drop Policy** Students will not be automatically dropped for non-attendance: if you are planning to drop the class or withdraw from the class, follow the instructions listed on registrar’s web site:
It is student’s responsibility to be familiar with the University Policy on dropping classes as described in the catalog and the TXSTATE website (see), to observe relevant deadlines, and to follow proper procedures for dropping classes.

**Incomplete Policy** Computer Science department has a strict policy regarding 'Incomplete grade'. It has to be approved by the chairman and thus an 'Incomplete grade' will only be granted under unexpected and truly severe situations, which must be supported by some official documents.

**Makeup Policy** Exercises and programming assignments **cannot** be made up. Midterm exam cannot be make up. Final exam may be made up in exceptional circumstances, with approval from the instructor.

**E-mail Policy**: During the work week, instructor will respond to personal emails within 24 hours. Instructor will review communication over the weekend but will respond on Monday to most situations. If you need to reach me by email, please use the subject line: Your Name, Course Name/Number, Topic. Please allow a full 24 hours before emailing me again about the same question or issue, and on Monday for inquiries sent over the weekend.

**Extra Credit Policy**: There is no extra credit or make up assignment available for this class.

**Plagiarism Policy**: Except where explicitly and specially allowed (such as group project), all work submitted in class is expected to be your individual work. Plagiarism will not be tolerated and if detected will result in an automatic ‘F’ grade. Please refer to [http://www.txstate.edu/effective/upps/upps-07-10-01.html](http://www.txstate.edu/effective/upps/upps-07-10-01.html) for Texas State’s Honor Code. • Do not include the code from the open source in your assignment (unless it is approved by the instructor) – penalty is 0 points for the assignment.

**Communication**

Best way to contact the instructor is to send her an email. All announcements, resources, and updated will be posted on TRACS [https://tracs.txstate.edu/](https://tracs.txstate.edu/) We will use the TRACS website for the following:

- Announcements (Announcement Tool)
- Assessments (Assessment tool) – In-class assessment is done using TRACs Assessment tool
- Grades (Gradebook tool)
- Programming assignment submissions (Assignments tool)
- Resources (Resources tool) will be grouped by topic number. Under each topic, you will find:
  - Lectures – pdf copies of lecture notes posted prior to the lecture
  - src – source code covered in lectures and useful examples for your programming assignments
  - videos – video capture of writing, compiling, and running the code in lecture

**GitHub**

Students will submit programming assignments through [https://git.txstate.edu](https://git.txstate.edu). Your login is your NetID. If you have questions about using git, please contact cs_helpdesk@txstate.edu.

**Accommodations for students with disability**

Any student requiring special accommodations, should inform me during the first two weeks of classes. The student should also contact the office of disability services at the LBJ student center. Students who qualify for extra time for exams must take their test with ATSD and must schedule their test at the same time the test is given in class.

**Academic Honor Code and Conduct**

You are expected to adhere to

- the University’s Academic Honor Code [http://www.txstate.edu/honorcodecouncil/Academic-Integrity.html](http://www.txstate.edu/honorcodecouncil/Academic-Integrity.html)
Course Schedule

Following is a tentative schedule for the class. Exact topics and dates may be updated as the course progresses.

<table>
<thead>
<tr>
<th>Lecture</th>
<th>Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan 22 01 Introduction</td>
<td></td>
</tr>
<tr>
<td>Jan 28 02 Java Programming 1</td>
<td></td>
</tr>
<tr>
<td>Jan 30 03 Java Programming 2</td>
<td></td>
</tr>
<tr>
<td>Feb 4 04 Java Programming 3</td>
<td></td>
</tr>
<tr>
<td>Feb 6 05 Java Exceptions</td>
<td>Feb 8 Assign 1 due</td>
</tr>
<tr>
<td>Feb 11 06 Object Oriented Design</td>
<td></td>
</tr>
<tr>
<td>Feb 13 07 Composition Inheritance</td>
<td></td>
</tr>
<tr>
<td>Feb 18 08 Polymorphism Abstraction</td>
<td></td>
</tr>
<tr>
<td>Feb 20 09 Java Collections Framework</td>
<td>Feb 19 Assign 2 due</td>
</tr>
<tr>
<td>Feb 25 10 Java Logging Framework</td>
<td></td>
</tr>
<tr>
<td>Feb 27 11 Java Unit Testing 1</td>
<td></td>
</tr>
<tr>
<td>Mar 4 12 Java Unit Testing 2</td>
<td></td>
</tr>
<tr>
<td>Mar 6 13 Review</td>
<td>Mar 8 Assign 3 due</td>
</tr>
<tr>
<td><strong>Mar 11 14 Midterm</strong></td>
<td></td>
</tr>
<tr>
<td>Mar 13 15 Java Logging</td>
<td>Mar 15 Peer reviews for Assign 3 due</td>
</tr>
<tr>
<td>Mar 25 16 Star UML</td>
<td></td>
</tr>
<tr>
<td>Mar 27 17 OOA UML Design</td>
<td></td>
</tr>
<tr>
<td>Apr 1 18 OOA UML Design and Analysis</td>
<td></td>
</tr>
<tr>
<td>Apr 3 19 OOA UML Analysis</td>
<td>Apr 5 Assign 4 due</td>
</tr>
<tr>
<td>Apr 8 20 Java Swing 1</td>
<td></td>
</tr>
<tr>
<td>Apr 10 21 Java Swing 2</td>
<td></td>
</tr>
<tr>
<td>Apr 15 22 Java Swing 3</td>
<td></td>
</tr>
<tr>
<td>Apr 17 23 Java Threads</td>
<td>Apr 19 Assign 5 due</td>
</tr>
<tr>
<td>Apr 22 24 Java Concurrency 1</td>
<td></td>
</tr>
<tr>
<td>Apr 24 25 Design Patterns 1</td>
<td></td>
</tr>
<tr>
<td>Apr 29 26 Design Patterns 2</td>
<td></td>
</tr>
<tr>
<td>May 1 27 Design Patterns 3 and JOM</td>
<td>May 3 Assign 6 due</td>
</tr>
<tr>
<td>May 6 28 Review and Research</td>
<td></td>
</tr>
<tr>
<td><strong>May 13 29 Final Exam 2-4:30 pm</strong></td>
<td></td>
</tr>
</tbody>
</table>