CS 3354: Object-Oriented Design and Programming  
Fall 2017

Instructor  Dr. Jelena Tešić (pronounced as Yeh-LE-nah TE-shich)  
Office Number  CMAL 307C  
Telephone (Office)  512.245.3436  
Email Address  jtesic@txstate.edu  
Faculty Profile  https://cs.txstate.edu/accounts/profiles/j_t463/  
Office Hours  Monday Wednesday 2:30 p.m. – 4 p.m. and by appt.

Section Information  CS3354-003  
Class Meetings  Monday Wednesday 5:00 p.m. - 6:20 p.m., DERR 240  
Open Labs  DERR 231 (Linux Lab) MCS 590 (Windows Lab)

Prerequisites  Grade of C or better in CS 2308

Course Material
- Class slides, notes, and source code posted on TRACS, No required textbook
  - Java: Chapter 1  
  - Inheritance & Polymorphism: Chapters 4.1-5 and 6.1  
  - Object Oriented Design Process: Chapters 2 and 3.4-5  
  - Design Patterns: Chapter 5.1-7  
  - Multithreading: Chapter 9  
- 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.
What is expected of student registered for CS 3354-003 in Fall 2017?

Students are expected to:

1. Attend instructional meetings
2. Do not distract or disrupt students during instructional meetings
3. Read emails from the instructor (sent to your TX State email).
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. Take midterm and final exam in the classroom during the exam time
9. Clearly communicate with the instructor regarding and 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>40%</td>
<td>Due date in TRACS and announced in the class</td>
</tr>
<tr>
<td>Midterm</td>
<td>25%</td>
<td>Wednesday Oct 18 5 p.m. (to be confirmed)</td>
</tr>
<tr>
<td>Final</td>
<td>30%</td>
<td>Friday Dec 8 5 p.m.</td>
</tr>
<tr>
<td>Quizzes and Class Participation</td>
<td>5%</td>
<td></td>
</tr>
</tbody>
</table>

Homework and Programming Assignments

The homework assignments involve drawing models and providing some written explanations. The programming assignments involve developing programs in Java.

Late assignments will incur 10% penalty per day, for up to 5 days. After the 5 days, no submission will be accepted.

Makeup Policy

Exercises and programming assignments cannot be made up. Exams may be made up in exceptional circumstances, with approval from the instructor.

TRACS URL: [https://tracs.txstate.edu/](https://tracs.txstate.edu/)

We will use the TRACS website for the following:

- Grades (Gradebook tool)
- Programming assignment submissions (Assignments tool)
- Lecture notes and Resources (code you can use in your programming assignments)
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.

Drop Policy

Students will not be automatically dropped for non-attendance: if you are planning to drop the class or withdraw from the class, you need to take action: http://www.registrar.txstate.edu/registration/dropping-or-withdrawing.html

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
  o the University’s Academic Honor Code http://www.txstate.edu/honorcodecouncil/Academic-Integrity.html

Do not list
  • Do not email your program to anyone except your partner (if approved by instructor) and
  • 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.
  • All assignments must be done either individually or in pairs or in groups, instructor will clearly specify the grouping for each assignment.

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
• Develop programs using multiple classes and composition.
• Describe the semantics of exception handling in Java, and use it to write reliable Java code.
• Read and write Java programs that use persistence (serializable objects).
• Read and write Java programs that use threads to implement concurrency.
• Describe and apply the Object-oriented design process to design a Java program.
• Read, design, and draw the following models using the Unified Modeling Language (UML): • Class diagrams • Sequence diagrams • State diagrams
• 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.
  o Adapter • Strategy • Command • Composite • Observer
• 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.