SYLLABUS: OBJECT-ORIENTED PROGRAMMING AND DESIGN PATTERNS

COURSE & INSTRUCTOR INFORMATION

Course

Course Title, Prefix, Number, Section: Object-Oriented Programming and Design Patterns, COSC 40453, 060

Semester and Year: Fall 20X

Number of Credits: 3

Course Component Type: Lecture

Instructor

Final Evaluative Exercise & Important Dates

FINAL: FRIDAY, DECEMBER 12, 2025
2:00 PM - 4:30 PM ~ REES-JONES 113

Note for students: The syllabus is your first course reading. It provides an orientation to, overview of the flow, and expectations of the course. You should turn to the syllabus for details on assignments and course policies.

Student Resources & Policy Information

Click or scan QR code for resources to support you as a TCU student. Please note section on Student Access and Accommodation, Academic Conduct & Course Materials Policies, and Emergency Response & TCU Alert.

COURSE DESCRIPTION

Catalog Description

This course teaches students to write code of high quality by providing in-depth coverage of object-oriented programming principles and techniques. Topics include (1)

fundamental object-oriented programming concepts (Objects and Classes, Abstraction, Encapsulation, Inheritance, Polymorphism, Interfaces, Abstract Classes, and Generics), (2) software design principles (Single Responsibility Principle, Open/Closed Principle, Liskov Substitution Principle, Interface Segregation Principle, Dependency Inversion Principle, Law of Demeter, Composition/Aggregate Reuse Principle), (3) design patterns (aka Gang of Four (GoF) patterns), (4) coding standards, (5) core features since Java 8. Each topic will be explained by using diagrammatic notations called Unified Modeling Language (UML) and numerous code demos.

Prerequisites & Concurrent Enrollment

Prerequisites: COSC 30403 with C- or better.

COURSE MATERIALS

Required Materials

None.

Supplementary Resources

- Effective Java 3rd Edition by Joshua Bloch
- Thinking in Java 4th Edition by Bruce Eckel
- Head First Design Patterns: A Brain-Friendly Guide 1st Edition by Eric Freeman, Bert Bates, Kathy Sierra, Elisabeth Robson

LEARNING OUTCOMES

Course Learning Outcomes

Students completing this course are expected to be able to:
1) Demonstrate adeptness of object-oriented programming in developing solutions to problems demonstrating usage of data abstraction, encapsulation, and inheritance.
2) Demonstrate ability to implement one or more patterns involving realization of an abstract interface and utilization of polymorphism in the solution of problems which can take advantage of dynamic dispatching.
3) Explain different OOP concepts, principles, design patterns, and methods.
4) Evaluate quality of code written by others.
5) Analyze and draw UML diagrams given a piece of open-source code written by others.
6) Learn and apply new Java 8 features in programs, including lambda expression, stream APIs etc.

COURSE REQUIREMENTS

Assignments

Exams

Midterm and final exam.

Homework

There are two homework assignments.

Project

There is an OOP project.

Grading Philosophy & Policy

Late Work

Every assignment must be uploaded to TCU Online no later than the beginning of class on the due day.

Late assignment incurs a 15% penalty for each late day, including weekend and holidays. Assignments that are late more than TWO days will **NOT** be accepted (except for Official University Absences or medical reasons).

Students are responsible for making sure the assignment is uploaded properly. Failing to do so results in a ZERO for that assignment.

Participation, Engagement & Attendance

Students are expected to attend all lectures.

Questions on Grading

Requests for re-evaluation of points on exams, assignments, and projects must be made to the instructor within one week of receiving your grade and accompanied by a brief written description of the grading error you believe was made. After this time, grades are final. Resubmission for re-evaluation subjects the entire assignment for review.

Course Assignments & Final Grade

Assignments	Percentage
Midterm	30
Final	30
Homework (2)	20
Project	20
Total	100

Grading Scale(s)

The following grading scale will be used in this course (overall score will be rounded up to the next whole number)

Grade	Score
A	90–100
B	80–89
C	70–79
D	60–69
F	0–59

COURSE SCHEDULE

This calendar represents my current plans and objectives. As we go through the semester, those plans may need to change to enhance the class learning opportunities. Such changes will be clearly communicated.

Week	Date	Topics
1	08/18	Introduction OOP Fundamentals
2	08/25	OOP Fundamentals (cont.)
3	09/01	OOP Fundamentals (cont.)
4	09/08	Java Reflection APIs Software Design Principles (SOLID)
5	09/15	SOLID (cont.)
6	09/22	SOLID (cont.)
7	09/29	Design Patterns - Singleton Design Patterns - Factory
8	10/06	Design Patterns - Builder Design Patterns - Prototype
9	10/13	Design Patterns - Proxy Design Patterns - Decorator
10	10/20	Design Patterns - Adapter Design Patterns - Façade Design Patterns - Chain of Responsibility
11	10/27	Design Patterns - Observer Design Patterns - Template Design Patterns - Strategy
12	11/03	Design Patterns - Iterator Design Patterns - Flyweight Design Patterns - State

Week	Date	Topics
13	11/10	New Java Features
14	11/17	New Java Features (cont.)
15	11/24	Thanksgiving
16	12/01	TBD
17	12/08	**Final Exam**: 2 PM - 4:30 PM Friday, December 12