

GRS CS 611:
Object Oriented Software Principles and Design (in Java)
Department of Computer Science
Boston University – Fall 2020

Description: An **advanced** course in Software Engineering with a specific focus on the design and development of software solutions using the Object Oriented Software paradigm.

This course is aimed on developing a deep understanding of the fundamental principles and techniques of object-oriented programming. Focus will be on the specification, programming, and analysis of **r**eliable, and **r**eusable software using object-oriented design principles. Topics include Classes, Objects and memory models, Abstraction, Inheritance and Polymorphism, Interfaces and Generic typing. Emphasis will be placed on using OO principles to design software against failure, and to design software for reusability through *well-known design patterns*.

All features of OO will be tied to Java specific implementations. We will study class definitions, designing a class inheritance structure, interfaces, generics, iterators and exception handling. A specific case study will be done to overlay the OO model on Java's built-in graphical user interface.

Prerequisites: Working knowledge of a programming language; Data Structures, SQL.

Goals and Objectives: The objective of the course is to provide the students with a strong foundation in object oriented principles and design, and to apply those skills in the context of an enterprise software development environment.

Instructor: Christine Papadakis-Kanaris, cpk@bu.edu, PSY 228B

Teaching Assistants: Athanasios Fillipidis (aflpd@bu.edu), Adam Streich (astreich@bu.edu), Jenny Li (jennjenn@bu.edu)

Meeting Times and Places

Lecture: T/Th 2:00 – 3:15 p.m. KCB 101

In keeping with the University's *Learn from Anywhere* (LFA) model, the class will run as follows:

- Lectures will run in a hybrid (in-person and live streamed) setting. Students are expected and encouraged to attend the lecture presentation, however, recorded lectures will be posted as soon as they are available.
- Discussion sessions will be offered in an exclusively remote setting and will begin on Wednesday, 9/16. Students have the option to attend weekly discussion sessions on an as needed basis. Students who are in a different time zone will have the ability to join a discussion section that is offered during a more convenient time for your time zone.
- **All** office hours will be offered remote only. There will be **no exceptions** for the TAs. Students who wish to have an in person meeting will have the opportunity to request an in person meeting with Professor Papadakis only.

Office Hours: Available on course Blackboard site

Course Materials

Textbooks: Program Development in Java, Abstraction, Specification, and Object-Oriented Design Barbara Liskov, John Guttag, Addison Wesley; Design Patterns: Elements of Reuseable Object-Oriented Software Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides Addison-Wessley; TBA

Optional: Introduction to Java Programming: Daniel Liang, Prentice Hall

Research Papers: as provided

Course Resources Blackboard (<https://learn.bu.edu>) will be used as a repository for the lecture notes from each class, for homework, and for general announcements. Piazza.com will be our tool for questions and discussion.

Requirements and Grading

Independent Assignments	35%
Midterm Exam	25%
• Written (40%)	
• Practicum (60%)	
Final Project	30%
Final Exam	10%

Independent Assignments: There will be (at least) three multi-part independent assignments throughout the course of the semester. The objective of each assignment is to put into practice the material and practices of OO design as presented in lecture. Your assignments will be graded in accordance to the mastery of the material as shown in practice. Completion of an assignment does not constitute mastery. Specifically, each assignment will be reviewed qualitatively on how well the principles of Object Oriented design have been put in practice.

Exams: There will be one comprehensive midterm exam and one final exam. The midterm exam will be in two parts and administered over two class periods: the first part will be an in class written exam focused on ensuring your understanding of Object Oriented Theory and Design. This part will account for 60% of your midterm grade. The second part will be an in class timed programming assignment. This part will account for 40% of your midterm grade.

The final exam will be a written qualitative analysis and critique of an object oriented design model that I will present to you. You will be assessed on your evaluation of the model and your suggestions at improving the design model to better reflect and adhere to OO principles.

Final Project: The course will culminate in a final project designed to **emulate a real-life development environment**. The objective for each student will be to investigate, understand and apply the software design process using object oriented constructs and techniques to a specific problem as presented by the instructor. Students will be randomly assigned to small development teams. Each team will be guided through the software development process, namely: analysis, specification and design, development, testing, and implementation. It is up to each team to use instructor resources to understand the system requirements.

Development teams will work independently and be encouraged to think creatively, but all teams must adhere to the problem objectives and specifications. Development will be done in the Java Programming language and the system developed is expected to provide an end to end solution, including use of local database and client graphical user interface. Students will be expected to **formally** present their design specification, project plan, and final solution.

Academic Misconduct

It is expected that you understand BU's Academic Conduct Code. The University and College take cheating very seriously. Cheating and plagiarism will not be tolerated in any course. Cases will be referred to the Dean's office and may result in loss of credit for an exam or assignment or other disciplinary action.

Please read the college's policy at: <http://www.bu.edu/academics/policies/academic-conduct-code>

Prohibited behaviors include:

- copying all or part of someone else's work, even if you subsequently modify it; this includes cases in which someone tells you what you should write for your solution
- viewing all or part of someone else's work (with the exception of work that is being done by

- other members of your group)
- showing all or part of your work to someone other than a peer in your assigned group.
- consulting solutions with those found online or in books
- posting your work where others (outside your group) can view it (e.g., online).

The following schedule is a guide to the organization of the material and assignments throughout the semester, but is subject to change at the discretion of the instructor.

Week	Weekly Topic Objective	Lecture Schedule	Assignment
Week 1	Foundational Overview Case Study	Course Introduction and Overview: Evolution of Computers, Languages, and Software Design	
Week 2	Understanding the Object Design Model: Applying the principles of <i>Abstraction</i> and <i>Encapsulation</i> .	The Java Programming Environment and model Designing for scalability using object decomposition vs. algorithmic decomposition Defining classes; The private interface; The public interface. Understanding the need for both.	Assignment #1, Part I: Basic Class design
Week 3	Understanding the Object Design Model: Applying the principles of <i>Inheritance</i> and <i>Polymorphism</i> .	Strong vs. Weak coupling; Designing for scalability through extendibility Applying the principles of <i>Inheritance</i> and <i>Polymorphism</i> . The binding process: static vs. dynamic.	Assignment #1, Part II: Designing an independent object model
Week 4	Understanding the Object Design Model: Abstraction by Specification. Understanding the ADT.	Interfaces, Java's alternative to multiple inheritance Abstraction vs. Interfaces; benefits and drawbacks.	Assignment #2, Part I: Designing for Modifiability

Week 5	Understanding the Object Design Model: Parametric Polymorphism: Using generic types to further abstraction Research Study	<i>Classes runs on Monday Schedule</i> Generic methods, interfaces and classes; Bounded vs. Unbounded types	
Week 6	Understanding the Object Design Model: Decomposition	Java Collection Library: <i>Iterators and Iterables, Comparators and Comparables.</i>	Assignment 2, Part II: Design for Usability, Reliability, and Correctness
Week 7	<i>Classes run on a Monday schedule</i>	Git Hub	
Week 8	Midterm exam	Written Exam Practicum Exam	
Week 9	Understanding Design Patterns	Behavioral Patterns (Iterator, Strategy) Behavioral Patterns(Observer)	Assignment #3, Part I: Designing for Extendibility
Week 10	Understanding Design Patterns	Creational Patterns (Singleton, Factory) Creational Patterns(Abstract Factory)	Pair Assignments
Week 11	Understanding Design Patterns	Structural Patterns (Composite, Adaptee, Proxy, Facade), Null Pattern	Assignment #3, Part II: Working together to evaluate an object design, test for extendibility.

Week 12	<p>Software Engineering:</p> <p><i>Excerpts from the Mythical Man Month:</i> <i>Aristocracy, Democracy and System Design</i></p> <p>Software Engineering: Understanding the Design Phase and clarity of objective</p> <p><i>Thanksgiving break</i></p>	<p>Event Driven Programming and GUI: Designing an independent object structure in a dependent environment.</p> <p>Understanding Software Development as an engineering discipline: Life Cycle, Quality Software, and the development team</p>	<p><i>Team Assignments:</i> The final project</p> <p>Importance of Group work, trust and communications.</p>
Week 13	<p>Software Engineering: Understanding the development phase and significance of team work and communication</p>	<p>Final Project: evaluating the presentations.</p> <p>Final Project: working lecture</p>	
Week 15	<p>Software Engineering: The development phase, making adjustments and polishing the presentation.</p>	<p>Finalizing schedule of presentations</p>	<p><i>Final Project Presentations</i></p>