Syllabus and overview

Dr. Safwan Qasem

King Saud University
College of Computer and Information Sciences
Computer Science Deptment

CSC 113: Course Description

This course continues the coverage of the fundamental concepts of Object Oriented Programming started in Programming I (CSC 111). In the Java Programming I, considerable time was spent concentrating on the fundamental programming concepts such as:

- Declarations of variables
- Basic control structures,
- Methods
- Arrays
- Classes & Objects.

CSC 113: Course Description

CSC 113 covers more advanced concepts and topics such as:

- Relationships between classes,
- Inheritance
- Polymorphism
- Abstract classes
- Error handling
- Files Input/Output
- Interfaces
- Data structures such as linked lists, stacks and queues
- Generics
- Graphical user interface.

Course Objectives

The objective of this course is to develop the students' ability to use the basics of object-oriented design and programming.

The students learn the characteristic features of object orientation – classes, methods, polymorphism, and inheritance – through both the lectures and a sequence of illustrative programming assignments.

Students will also be introduced to data structures, generic programming, event driven programming and graphical user interface tools.

Course Learning Outcomes

Upon completing CSC113, students should have the following capabilities:

1. *Understand classes and instances, and how programs can be designed as a collection of communicating objects.*
2. *Understand and be able to design and implement programs using object oriented programming concepts like: encapsulation, inheritance, polymorphism, abstract classes and methods, and Interfaces.*

Course Learning Outcomes

3. *Use standard documentation, such as UML class diagrams and online Java documentation.*
4. *Learn how to compile programs on at least one platform using command lines and / or IDE.*
5. *Create and manipulate dynamic data structures, such us linked lists, stacks and queues.*
6. *Create and access files with Java.*
7. *Design and Implement event-driven interactive programs.*

Course Learning Outcomes

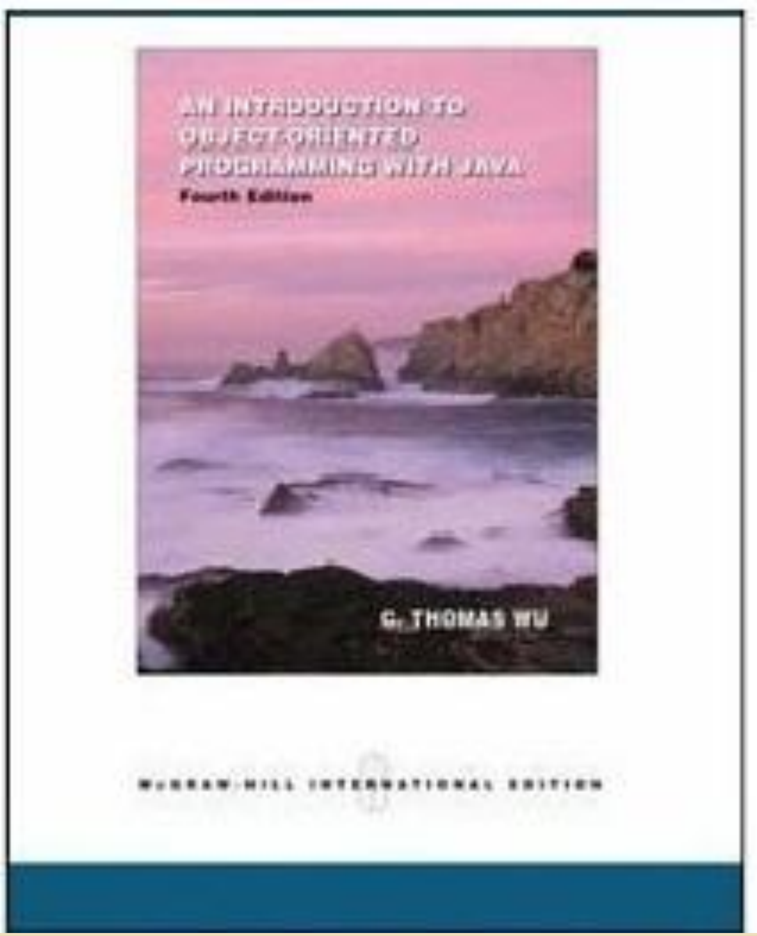
8. *Students should learn how to work in groups towards achieving the same goal.*

Outcomes will be assessed using classroom and lab performance, graded homework assignments (lecture and lab), quizzes (lecture and lab), graded lab exercises, course project (lab), and midterm and final examinations (lecture and lab).

Anti-cheating policy: Any student who cheat or copy from each other will earn a ZERO and go to the department chairman office to explain their behavior.

Course Topics

- Array of Objects
- methods with object arguments and returning objects
- Relationship between Classes using UML
- Inheritance, polymorphism and Interface
- Graphical User Interface
- Exception handling
- File Input/output (Binary, Text and Object files)
- Data structures, Linked Lists
- Data structures, Stacks, Queues
- Generics



Text Book

"An Introduction To Object-Oriented Programming With JAVA", 4th Edition, 2006

Author: C. Thomas WU

Editor: McGraw-Hill Higher Education

ISBN: 0-07-111680-X

Expected Performance Criteria

The student is expected to:

- □ *Attend lecture, tutorial and lab classes*
- □ *Perform assigned programming tasks in Java.*
- □ *Submit homework assignments on time.*
- □ *Pass two midterm examinations.*
- □ *Pass a final Lab exam.*
- □ *Pass the final written exam.*
- □ *Achieve an average mark of at least 60%.*

Grading

Final Exam:	40%
Midterm 1:	15%
Midterm 2:	15%
Lab:	10%
Homework:	10%
Quizzes:	5%
Project:	5%