King Saud University

College of Computer and Information Sciences

Department of Computer Science

CSC113 Computer Programming II

(3-2-1) - Required Course

Catalogue Description

This course continues the coverage of the fundamental concepts of Object Oriented Programming started in Programming I (CSC 111). It covers more advanced concepts and topics such as relationships between classes, inheritance, polymorphism, abstract classes, error handling, interfaces, generics and data structures such as linked lists, stacks and queues, in addition to graphical user interface.

http://faculty.ksu.edu.sa/sqasem/Pages/CSC113ProgrammingLanguagesII.aspx

Pre-requisites:

CSC 111 computer Programming I

Prerequisite to:

CSC 212 Data Structures
CSC 261 Artificial Intelligence Programming Languages

Textbook:

An Introduction To Object-Oriented Programming With JAVA (Fourth Edition), C. Thomas WU, 2006, McGraw-Hill Higher Education, ISBN 0-07-111680-X

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 study list data structures, 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.*
3. *Use standard documentation, such as UML class diagrams and online Java documentation.*
4. *Create and manipulate dynamic data structures, such us linked lists, stacks and queues.*
5. *Create and access files with Java.*
6. *Design and Implement event-driven interactive programs.*
7. *Students should learn how to work in groups towards achieving the same goal.*
8. *Learn how to compile programs on at least one platform using command lines and / or IDE.*

Expected Performance Criteria

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).

The student is expected to perform assigned programming tasks in Java and to pass three written examinations.

Topics:

1. Array of Objects
2. methods returning objects

3. Relationship between Classes using UML
4. Inheritance, polymorphism and Interface
5. Graphical User Interface
6. Exception handling
7. File Input/output (Binary, Text and Object files)
8. Data structures, Linked Lists, Stacks, Queues
9. Generics

Schedule:

15 weeks of 3 50-minutes lectures, two hours of Lab and one-hour tutorial.

Relationship of course to ABET Criteria:

Criterion 2 - Program Educational Objectives:

This course allows the student to gain the necessary skill and experience to contribute to either a research or development project involving the design concepts of modern uni-processor computers

Criterion 3 – Program Outcomes:

a- an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline
Students write object oriented and event driven programs.

b- an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
----

c- an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired goals.
Students are requested to design small systems and implement and test their solutions in HW and Lab assignments.

d- an ability to function effectively on teams to accomplish a common goal
Students work by groups of two to accomplish one or more mini-projects during the course.-

e- an understanding of professional, ethical, legal and social issues and responsibilities
----

f- an ability to communicate effectively
----

g- an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues
----

h- ***a recognition of the need for, and an ability to engage continuing professional development***
----

i- an ability to use the current techniques, skills, and tools necessary for computing practice.
The students learn the current dominant paradigm, object-oriented design and programming and learn how to use Integrated Development Environment and compile programs using command lines.

j- an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.
----

k- an ability to apply design and development principles in the construction of software systems of varying complexity
----

Course Learning Outcomes	***(a)***	***(c)***	***(d)***	***(i)***
	50%	***25%***	***10%***	***15%***
1	***X***			
2		***X***		
3	***X***			
4	***X***			
5	***X***			
6		***X***		
7			***X***	
8				***X***

Criterion 4 – Professional Component:

This course provides the students with basic design and programming skills that will be used (and further developed) throughout the curriculum.

Prepared by: Dr Safwan Qasem

Reviewed by: Dr. Salah Hammami

Approved by :