CPCS-203 Syllabus

Catalog Description

CPCS-203 Programming (II)
Credit: 3 (Theory: 2, Lab: 0, Practical: 1)
Prerequisite: CPCS-202
Classification: College Required

The objective of this course is to cover the basic and intermediate concepts of object oriented programming. Topics include the use of numerical data, primitive data types, selection statements, repetition statements, class, standard classes, object, instance of a class, object oriented programming, software development, constructor, methods, call by values, call by reference, UML, state memory diagram, class diagram, arrays, method overloading, constructor overloading, overriding, this, new, exception and assertions, Inheritance, encapsulation, data abstraction, private, public, protected, exception, inheritance, abstract class, polymorphism, file handling.

Class Schedule

Meet 50 minutes 3 times/week or 80 minutes 2 times/week

Lab/Tutorial 90 minutes 1 times/week

Textbook

Dr. Liang, , "Introduction to Java Programming, Comprehensive Version 9th Edition", Fonenix inc; 9 edition (2012-04-07)

ISBN-13 9780132936521 **ISBN-10** 0132936526

Grade Distribution

Week	Assessment	Grade %
4	Homework Assignments 1	6
4	Quiz 1	1
6	Quiz 2	1
6	Homework Assignments 2	6
7	Exam 1	10
8	Lab Exam 1	5
9	Quiz 3	1
9	Homework Assignments 3	6
11	Homework Assignments 4	6
11	Quiz 4	1
12	Exam 2	15
14	Quiz 5	1
14	Homework Assignments 5	6
15	Lab Exam 2	10
16	Comprehensive Final Exam	25

Last Articulated

December 11, 2017

Relationship to Student Outcomes

a	b	c	d	e	f	g	h	i	j	k	l	m	n
x	x	x											

Course Learning Outcomes (CLO)

By completion of the course the students should be able to

1. Apply String manipulation using java programming language (a)
2. Apply File Input and Output operations using java programming language (a)
3. **Apply the fundamental concepts of object identity, type constructors, encapsulation, and information hiding, that underlie object oriented approaches to software development (b)**
4. Illustrate Classes and Objects, and how a program can be designed as a collection of communicating objects (a)
5. Construct the UML diagrams that depict the Java program structure (c)
6. **Analyze a real-world situation in an object-oriented way by interpreting the factors that contribute to a good object oriented solution (b)**
7. **Demonstrate the fundamental and relationship between object-oriented inheritance and subclasses (b)**
8. Apply object-oriented principles of polymorphism using real life examples (c)
9. Construct an arrayList and apply storing, retrieving and manipulating object in an ArrayList (c)
10. Apply interfaces and abstract classes to extend the functionality of classes (b)
11. **Design classes and class hierarchies from problem statements using sub classing, abstract classes, and interfaces to achieve polymorphism in object oriented software (c)**
12. Describe the benefit of generic classes (a)
13. **Apply exception handling mechanisms in Java to make programs more robust (b)**
14. Design simple GUI interfaces for a computer program to interact with users and handle events driven (c)

Coordinator(s)

Dr. Kawthar Moria, Associate Professor

Dr. Asif Irshad Khan, Assistant Professor

CPCS-203 Syllabus

Topics Coverage Durations

Topics	Weeks
Review: Multidimensional Arrays	1
Strings	1
Text I/O	1
Objects and Classes	2
Thinking in Objects	1
Inheritance and Polymorphism	3
Abstract Classes and Interfaces	1
Generics	1
Exception Handling	2
JavaFx UI controls	1
Event Driven Programming	1