

## COURSE DESCRIPTION

<b>Department and Course Number</b>	CS 201	<b>Course Coordinator</b>	Behzad Parviz
<b>Course Title</b>	Introduction to Programming	<b>Total Credits</b>	5

### Current Catalog Description:

Algorithm development for structured programming and computer programming; designing, coding, debugging, and documenting programs. Laboratory activities on problem analysis and software development.

### Textbook:

Deitel and Deitel., *Java How to Program, Sixth Edition*, Prentice Hall, 2004.

Deitel and Deitel., *Java in the Lab, Sixth Edition*, Prentice Hall, 2004.

### References:

Liang, Daniel., *Introduction to Java Programming, 4<sup>th</sup> Edition*, Prentice Hall, 2002.

### Course Goals:

At the end of the course, students

- Understand the basic programming concepts.
- Are able to create simple classes with a few methods.
- Are able to divide a problem into its logical components.
- Understand how good program design reduces coding and debugging time.
- Are able to design and code mid-level problems.

These course goals contribute to the success of **Student Learning Outcomes 1.a, 1.d, 1.e, 5, and 6.**

### Prerequisites by Topic:

Algebra and Trigonometry.

### Major Topics Covered in the Course:

- Introduction to Computers and the Internet
  - What is a Computer
  - Computer Organization
  - History of Java
  - The Internet and the World Wide Web
  - Basics of a typical Java Environment

- Introduction to Java applications
  - Studying a Simple Program in Java
  - Displaying Text in a Dialog Box
  - Memory Concepts
  - Arithmetic
  - Decision Making: Equality and Relational Operators
  - Programming Design
  
- Introduction to Classes and Objects
  - Classes, Objects, Methods and instance variables
  - Declaring a Class with a method and Instantiating an Object of a Class
  - Declaring a Method with a Parameter
  - Primitive Types vs. Reference Types
  - Initializing Objects with Constructors
  
- Control Statements
  - Algorithms
  - Pseudo-code
  - if...else* Selection Statement
  - while* Repetition Statement
  - Formulating Algorithms
  - Compound Assignment Operators
  - Primitive Types
  - for* Repetition Statement
  - do...while* Repetition Statement
  - switch* Multiple-Selection Statement
  - break* and *continue* Statements
  - Logical Operators
  
- Methods
  - Program Modules in Java
  - static* Methods, *static* Fields
  - Declaring and using Methods with Multiple Parameters
  - Argument Promotion and Casting
  - Java API Packages
  - Scope of Declarations
  - Method Overloading
  
- Arrays
  - Declaring and Creating Arrays
  - Examples Using Arrays
  - Passing Arrays to Methods
  - Multidimensional Arrays
  - Variable-Length Argument Lists

**Laboratory Projects (specify number of weeks on each):**

Each week students complete a 3-hour lab project on a selected. In addition, they design and develop three to five large projects covering the materials taught in the course.

- Week 1: Program development using basic programming concepts and constructs.
- Week 2: Program design and Programming applications in Java using Jbuilder and JDK tools.
- Week 3: Problem development process using pseudo-code.
- Week 4: Program development using classes, and objects.
- Week 5: Program development using simple control statements for decision making and repetition: *if...else* and *while* statements.
- Week 6: Program development using control statements: *for*, *do...while*, *switch*, *break*, and *continue* statements.
- Week 7: A deeper look inside objects. Program development using class-library methods and user defined methods.
- Week 8: Program development emphasizing modularity.
- Week 9: Program development using array of items of the same types.
- Week 10: Project design and development using the main topics covered in the course.

**Estimate Curriculum Category Content (Quarter Hours)**

Area	Core	Advanced	Area	Core	Advanced
Algorithms	.75		Data Structures	1.5	
Software Design	.75		Prog. Languages	2.0	
Comp. Arch.					

**Oral and Written Communications:**

Written documentation of software built in labs and homework assignments.

**Social and Ethical Issues:**

No significant component.

**Theoretical Content:**

Logic design and algorithm development.

**Problem Analysis:**

In this course, students learn the basic concepts of the Java programming language. They learn algorithm development for structured programming, designing, coding, debugging, and documenting programs.

**Solution Design:**

Solution design in this course mostly involves generating pseudo-code for program development. Students also learn how good program design reduces coding and debugging time.