

## COURSE DESCRIPTION

<b>Department and Course Number</b>	CS 203	<b>Course Coordinator</b>	Vladimir Akis
<b>Course Title</b>	Programming with Data Structures	<b>Total Credits</b>	5

### Current Catalog Description:

Advanced programming techniques; elementary data structures such as dynamic arrays, linked lists, stacks, queues, and trees, sorting and searching algorithms. Laboratory activities on problem analysis and software development.

### Textbook:

Deitel and Deitel., *Java How to Program, 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 are able to

- Design recursive methods
- Know the standard searching and sorting algorithms and their efficiency
- Use dynamic data structures.
- Know the standard Abstract Data Types, and their implementations.
- Use Java Generics and Collection.

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

### Prerequisites by Topic:

Object Oriented Programming and graphical user interfaces.

### Major Topics Covered in the Course:

- Recursion
- Searching and Sorting
- Data Structures
  - Linked Lists
  - Stacks
  - Queues
  - Trees
- Generics
- Collections

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

Each week the students complete 3-hour lab projects on a selected topic. A typical lab schedule of weekly topics is:

- Week 1: Programming with Recursion.
- Week 2: Programming using Searching Algorithms.
- Week 3: Programming using Sorting Algorithms.
- Week 4: Programming using Linked Lists.
- Week 5: More Programming using Linked Lists.
- Week 6: Programming with Stacks and Queues.
- Week 7: Programming with Binary Trees. Implementation of Recursive Search, Insert, Delete algorithms.
- Week 8: Programming using Trees.
- Week 9: Programming using Generics and Collections.
- Week 10: More Programming using Generics and Collections.

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

Area	Core	Advanced	Area	Core	Advanced
Algorithms	.75		Data Structures	.75	
Software Design	1.5		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:**

- Recursion
- Efficiency of Algorithms
- Abstract Data Types

**Problem Analysis:**

In the first part of the course, students learn the concept of recursion and how to design recursive methods. Next they are introduced to searching and sorting algorithms, both sequential and recursive. They learn how to calculate the efficiency of these algorithms using big-O notation. They learn to implement the standard data structures and their algorithms. In the latter part of the course, students learn how to use Generics and Collections (the data structure classes provided by Java library, such as Arrays, ArrayList, Iterator, LinkedList, Vector, Stack, etc.,).

**Solution Design:**

Solution design in this course mostly involves the use of Data Structures and Algorithms appropriate for certain programming tasks and the development of efficient large programs.