



## College of Arts and Applied Sciences

---

### Course Syllabus

**Course Number:** CMPS 160

**Course Title:** Data Structures

**Course instructor/Instructors:** Dr. Nafaâ Jabeur

**Lecture Time:** Saturday, Monday, and Wednesday from 09:00 to 09:50 (Section 1)

Saturday, Monday, and Wednesday from 11:00 to 11:50 (Section 1)

**Office Hours:** Saturday: 10:00 to 10:50, Sunday: 09:30 to 10:30, Monday: 14:00 to 15:00

**Email:** [nafaa\\_jabeur@du.edu.om](mailto:nafaa_jabeur@du.edu.om)

**Office Telephone Extension:** 7210

**Semester:** Spring 2011

**Classroom:** CAAS-003B (Section 1) and CAAS-101C (Section 2)

---

### APPROACH

**Description:**

This course reviews the main concept seen in CMPS 110 and promotes the design and the writing of medium/large programs. It includes detailed studies of data structures and data abstraction such as queues, linked lists, and trees. In addition, it presents some of the existing sorting algorithms.

**Learning Outcomes:**

Students are expected to demonstrate the following knowledge, skills, and competencies:

**Knowledge:**

- Being familiar with Object-Oriented Programming
- Having experience with the Java language
- Understanding the concepts and principles of the different data structures, including their design and implementation issues.
- Understanding the concept of Recursion

**Cognitive Skills:**

- Analyzing problems and designing solutions
- Constructing programming codes through designing, testing, and executing them in the actual environment
- Selecting appropriate data structures
- Gaining experience in developing Java applications

**General Competencies:**

- Working with team members of a development group
- Presenting sound arguments

**DEPLOYMENT**

**Content:**

Topic	Number of hours
A Review of Some Basic Java Features	9
Object-Oriented programming	6
Linked Data Representations	6
Recursion	6
Stacks and Queues	6
Sorting	6
Introduction to Trees and Graphs	3
Introduction to Hashing and the Table ADT	3

**Learning/Teaching Strategies:**

- Discuss the important aspects of programming languages.
- Reshape the plan of the course depending on whether students are having difficulties with certain topics. I will be more in tune with students' backgrounds.
- Invite students to read the material before coming to class.
- Strongly encourage collaboration to solve some proposed exercises.
- Plan assignments during which collaboration is not allowed.
- Encourage students to do self-work. This course gives to students the required background for coding in Java. However, students must work outside the course in order to improve their skills.
- Encourage students to take an active role in their learning in this course. This includes regular attendance, paying attention in class, reading the slides, and completing all course requirements. Students are encouraged to study with their classmates outside of class. Programming assignments usually require a lot more time than expected, so starting early and working almost every day is necessary.
- Motivate students to review the materials, participate in class, and compete by frequently asking questions and giving extra marks for correct answers.

**Reading Materials and References:**

Textbook: Thomas A. Standish, **Data structures in Java**, Addison Wesley Longman Inc. Edition, 2000 (available at DU library and references that may be found elsewhere)

**References**

Nafaâ Jabeur, **Data Abstraction - Handouts**, 2010

David Eck, **Introduction to programming using Java**, Version 5, 2007 (available for free download at: <http://www.lulu.com/content/612392>)

John Lewis and William Lotfus, **Java Software Solutions: Foundations of Program Design**, Addison Wesley Longman Inc. 5<sup>th</sup> Edition, 2007

Nafaâ Jabeur, **Introduction to Programming - Handouts**, 2009

Handouts: Handout will be provided for each module

Software: Java will be used as the programming language.

Websites: <http://www.du.edu.om/caas/cs/nafaa/courses.html>

Assignments: 2 theoretical assignments are scheduled for this course.

Programming assignments: 2 assignments will be given during the term. Solutions to the assignments will be made available soon after assignments are collected so no late assignment will be accepted.

Tests: 4 tests targeting comprehension and programming skills will be carried out during the semester. The idea is to push students to review what has been taught more frequently. Only, the 3 best grades got by a student will be counted.

Final exam: A final exam will be given at the end of the semester

**Grading Criteria:**

Attendance/Participation	5%
Assignments	30%
Tests	30%
Final Exam	35%

**Important Dates:**

Tests: **at the end of every chapter (almost every 2/3 weeks)**

Final Exam: **will be scheduled by the University**

Last day of withdrawal from the course with a "W" grade is: **June 01 2011 - Wednesday**

**RESULTS**

**Evidence of Learning:**

The evidence of learning will be demonstrated by the scores on the exams, the quality of papers, presentations, and projects.

**IMPROVEMENTS**

- Course material is improved periodically.
- Emphasize on practical aspects to support theoretical concepts.

**Academic Honesty**

Plagiarism is using other people's work without proper referencing. Students are expected to complete all work with the highest standard of honesty and integrity. Plagiarism, forgery, cheating or any form of academic misconduct will not be tolerated. Any of the above may cause a student's final course grade to be lowered significantly or the student may receive a failing grade, depending on the severity of the offence.