

## Programming Methodology and Abstractions - Course Syllabus

---

**Course Number:** CS 207

**Course Title:** Programming Methodology and Abstractions

**Academic Semester:** Summer                      **Academic Year:** 2015/ 2016  
**Semester Start Date:** Jun 05, 2016              **Semester End Date:** Aug 04,2016

**Class Schedule:** Lectures: Monday - Wednesday, 9:00 am to 12:00 pm. Additional lab session schedule TBD according to TA availability

**Instructor(s) Name(s):** Malek Smaoui  
**Email:** malek.smaoui@kaust.edu.sa

**Office Location:** Bldg. 1, flr. 4, rm. 4124  
**Office Hours:** By appointment

**Teaching Assistant name:** TBD  
**Email:**

### COURSE DESCRIPTION FROM PROGRAM GUIDE

Computer programming and the use of abstractions. Object-oriented programming, fundamental data structures (such as stacks, queues, sets) and data-directed design. Recursion and recursive data structures (linked lists, trees, graphs). Introduction to basic time and space complexity analysis. The course teaches the mechanics of the C, C++ or Java language as well as an example of media library

### COMPREHENSIVE COURSE DESCRIPTION

The course teaches the mechanics of the C, C++ to introduce and reinforce:

- Computer programming and the use of abstractions.
- Software engineering principles of data abstraction and modularity.
- Object-oriented programming and encapsulation of fundamental data structures.
- Recursion and recursive data structures.
- Basic time and space complexity analysis.

## GOALS AND OBJECTIVES

At the end of the course, student should be able to:

- solve problems by writing computer programs
- solve problems iteratively and recursively
- design both structured and object-oriented programs
- use basic data structures available in standard libraries
- design custom data structures
- evaluate the running time and space usage of a program
- use third party libraries to compose a computer program of significant size and value.

## REQUIRED KNOWLEDGE

Basic calculus knowledge

## REFERENCE TEXTS

Programming Abstractions in C++, Eric Roberts, Prentice Hall, 2013.

## METHOD OF EVALUATION

Percentages %	Graded content
30%	4 programming assignments
15%	3-5 pop paper quizzes
15%	Programming midterm exam
20%	Programming project
20%	Paper final exam

## COURSE REQUIREMENTS

### Assignments

Assignments are sets of short to medium length programming exercises, due within one week to cover 1-2 weeks of material.

Midterm consists in solving 1-2 exercises in the lab in 1-1.5 hours.

Project is assigned in the last 2-3 weeks and should use most of what has been learned.

### Course Policies

Assignments, midterm and project are to be submitted via blackboard.

## NOTE

The instructor reserves the right to make changes to this syllabus as necessary.