



Philadelphia University
Faculty of Engineering
Department of Computer Engineering
First semester, 2015/2016

Course Syllabus

Course Title: : Algorithms and Data Structures	Course code: 630224
Course Level: 3 rd	Course prerequisite (s) and/or co requisite (s): Object Oriented Programming (630221)
Class Time: 9:45-11:15 Mon., Wed.	Credit hours: 3

Academic Staff

Specifics

Name	Rank	Office Number and Location	Office Hours	E-mail Address
Dr. laheeb Al-Zubeidy	Associate Prof.	711	14 :00- 16 :00	laheeb_alzubaidy@yahoo.com

Course module description:

This course introduces the fundamentals of structuring and manipulating data: sorting, searching, recursion, lists, stacks, queues, trees, graphs, tables. Introduction to the analysis of algorithms. Advancement in C++ skills and techniques.

Course module objectives:

- ❖ Understand algorithms, time complexity and space calculating
- ❖ Understand the sorting and searching fundamentals.
- ❖ Describe and /or define the Abstract Data Types; including lists, stacks, queues, trees, hash tables and graph.
- ❖ Understand, explain, demonstrate, and evaluate alternate implementations of examples of the methods associated with Abstract Data Types.
- ❖ Implement and test Abstract Data Types in generic programs using C++.

Course/ module components

- ❖ Data Structures and Algorithms in C++, Second Edition ,By Adam Drozdek,2000
- ❖ Books (title , author (s), publisher, year of publication)
C++ How to Program, 3rd By Deitel & Deitel, prentice-Hall, 2001, ISBN: 0-13-089571-7
- ❖ Support material (s) (vcs, acs, etc).
- ❖ Study guide (s) (if applicable)
- ❖ Homework and laboratory guide (s) if (applicable).

Teaching methods:

Duration: 16 weeks, 48 hours in total

Lectures: 32 hours, 3 per week

Homework: 7-8 homework assignments

Learning outcomes:

- Knowledge and understanding
 - ❖ Have a clear understanding of the Data Abstraction term.
 - ❖ Have a good knowledge of what comprises a correct program in C++.
 - ❖ Have knowledge of design guidelines.
- Cognitive skills (thinking and analysis).
 - ❖ Be able to design, code, and test C++ programs, which meet requirements expressed in English.
 - ❖ Have knowledge of design guidelines.
 - ❖ Be able to write algorithms for solving problems.
- Communication skills (personal and academic).
 - ❖ Be able to understand the documentation for, and make use of, the C++ library.
 - ❖ Be able to write a C++ program.
 - ❖ Be able to design, code, and test C++ programs, which meet requirements expressed in English.
- Practical and subject specific skills (Transferable Skills).
 - ❖ write computer programs to solve practical engineering problems
 - ❖ Design efficient computer programs to solve practical engineering problems

Course Intended Learning Outcomes									
A - Knowledge and Understanding									
A1.	A2.	A3.	A4.	A5.	A6.	A7.	A8.		
B - Intellectual Skills									
B1.	B2.	B3.	B4.	B5.	B6.	B7.	B8.	B9.	
C - Practical Skills									
C1.	C2.	C3.	C4.	C5.	C6.	C7.	C8.	C9.	C10.
D - Transferable Skills									
D1.	D2.	D3.	D4.	D5.	D6.	D7.			

Assessment instruments

- Short reports and/ or presentations, and/ or Short research projects
- Quizzes.
- Assignments.
- Final examination: 50 marks

<u>Allocation of Marks</u>	
Assessment Instruments	Mark
First examination	20
Second examination	20
Final examination: 50 marks	40
Reports, research projects, Quizzes, Assignments, Projects	20
Total	100

Documentation and academic honesty

- Documentation style (with illustrative examples)
- Protection by copyright (N/A)
- Avoiding plagiarism. (According to the university rules and regulations)

Course/module academic calendar

week	Basic and support material to be covered	Homework/reports and their due dates
(1)	Programs complexity	
(2)	Sorting and Searching algorithms	Building Several Functions to represent several sorting and searching algorithms
(3)	Data representation : linear and linked lists	Building linear linked list
(4)	Arrays & matrices	
(5)	Linear Stacks	Building linear Stack with some extra functions
(6) First examination	First examination	18-26\11\2015
(7)	Queue	Building linear and Dynamic Queue and used it with a given applications
(8)	Dynamic Stacks	Building Dynamic stack and calculate time complexity for some methods.
(9)	Priorities queue	Building a dynamic Priorities Queue.
(10)	Binary tree & usual Tree	
(11)	Algorithm of converting trees	27\12\2015-5\1\2016
(12)		Building an application using polymorphism
(13)		
(14)	graphs	Build a Graph and applied some searching algorithm to travels through all nodes
(15) Specimen examination (Optional)	Hash tables	Building and implementing Hash tables
(16) Final Examination		30\1-7\2\2016

Expected workload:

On average students need to spend 2 hours of study and preparation for each 50-minute lecture/tutorial.

Attendance policy:

Absence from lectures and/or tutorials shall not exceed 15%. Students who exceed the 15% limit without a medical or emergency excuse acceptable to and approved by the Dean of the relevant college/faculty shall not be allowed to take the final examination and shall receive a mark of zero for the course. If the excuse is approved by the Dean, the student shall be considered to have withdrawn from the course.

Module references**Books**

- ❖ Data Structures Using C++, 2nd Edition, D.S. Malik
- ❖ Data Structures and Problem Solving Using C++, 2nd Edition, M. A. Weiss
- ❖ Algorithms and Data Structures in C++ . By Leendert Ammeraal. 1996.
- ❖ C++ How to program .By H.M.Deitel & P.J.Deitel. 2 ed , Prentice- hill, 1998.
- ❖ Data Structures and Algorithms in C++ ,1ST edition , by Michael T. Godrich, Roberto Tamassia, David M. Mount Michael T. Goodrich Wiley , 2002.

Journals

Websites
