CPCS-223 Syllabus

Catalog Description

CPCS-223 Analysis and Design of Algorithms
Credit: 3 (Theory: 3, Lab: 0, Practical: 1)
Prerequisite: CPCS-204
Classification: Department Required

This is the first of a two-course sequence on computer-based algorithmic solution design and advanced data structures. It introduces fundamental algorithms for classic computing problems, the techniques used to construct them, their performance and applications. Topics include: formal definition and characterization of algorithms, recurrence relations, asymptotic notation and efficiency classes, iterative and recursive algorithm efficiency, empirical analysis of performance, advanced data structures such as balanced trees, and the design techniques: brute force, divide-conquer, decrease-conquer, problem transformation, and trading space for time.

Class Schedule

Lab/Tutorial 90 minutes 1 times/week

Meet 50 minutes 3 times/week or 80 minutes 2 times/week

Textbook

Anany Levitin, , "Introduction to the Design & Analysis of Algorithms", Addison-Wesley Longman; 3 edition (2012)

ISBN-13 9780132316811 **ISBN-10** 0132316811

Grade Distribution

Week	Assessment	Grade %
6	Exam 1	15
10	Project (Individual)	15
12	Exam 2	15
15	Homework Assignments	15
16	Comprehensive Final Exam	40

Last Articulated

April 2, 2018

Relationship to Student Outcomes

a	b	c	d	e	f	g	h	i	j	k
x	x	x					x			

Course Learning Outcomes (CLO)

By completion of the course the students should be able to

1. Discuss the formal definition of algorithm. (a)
2. Discus how the efficiency of algorithms is characterized and measured. (a)
3. Solve simple recurrence relations using select (covered) techniques. (a)
4. Determine the time efficiency of simple iterative or recursive algorithm given its pseudocode using select (covered) techniques and theorems. (b)
5. **Use given algorithm to solve small instances of select (covered) problems by hand. (b)**
6. Write proper (well-formed) pseudocode based on knowledge of algorithm operation. (Intent: not a design outcome). (a)
7. Analyze select (covered) decrease-conquer algorithms given a pesudocode. (b)
8. **Compare select (covered) algorithms and related data structures in terms of design, operation, performance, and applications. (b)**
9. Analyze select (covered) divide-conquer algorithms given a pesudocode. (b)
10. [x] MERGED WITH 8 - Compare select (covered) algorithms and related data structures in terms of operation, performance, and applications. (b)
11. **Analyze select (covered) transform-conquer algorithms given a pesudocode. (b)**
12. Compare select (covered) data structures with respect to dictionary operations. (b)
13. **Apply empirical techniques to assess and report the performance of one or more algorithms. (h)**
14. Implement algorithm given its pseudocode in programming language and environment of choice. (a)
15. **Design an algorithm to solve a problem based on requirements or solution descriptions, or an outline of steps. (c)**

Coordinator(s)

Dr. Muhammad Al-Hashimi, Assistant Professor

CPCS-223 Syllabus

Topics Coverage Durations

Topics	Weeks
Introduction to algorithms	1
Fundamentals of algorithm efficiency	2
Brute-force algorithms	2
Decrease-conquer algorithms	2
Emprical evaluation of performance	1
Divide-conquer algorithms	3
Problem transformation algorithms and data structures	3
Trading space for time	1