

Detailed Course Description - Course Plan Development and Updating Procedures/ Computer Information Systems Department	QF01/0408-3.0E
---	-----------------------

Faculty	Faculty of Science and Information Technology	Department	Computer Information Systems
Course Number	0113313	Course Title	Algorithms
Number of Credit Hours	3	Pre-Requisite/Co-Requisite	Data Structure

This course concerns with: fundamentals of algorithmic problem solving, algorithm design techniques, and methods of specifying and coding an algorithm; Fundamentals of the analysis of algorithm efficiency and efficiency levels of algorithms and formal definitions of (O -notation, Ω -notation and Θ -notation); Concepts of sorting such as: selection sort, bubble sort, merge sort, quick sort and heap sort; Concepts of sequential search, depth-first search, breadth-first search, binary search; concepts of hashing such as: open hashing and closed hashing and B-trees; Concepts of Greedy techniques such as Prim's, Kruskal's, Dijkstra's Algorithm and Huffman trees and codes; Concepts of limitations of algorithm power, decision trees for searching a sorted array, P , NP , and NP -Complete Problems.

Course Goals And Learning Outcomes	
Goal 1	Fundamentals of algorithmic problem solving
Learning Outcomes	1.1 Learning algorithm design techniques, designing data structures, methods of specifying and coding an algorithm.
Goal 2	Fundamentals of the Analysis of Algorithm Efficiency
Learning Outcomes	2.1 Learning the efficiency levels of algorithms and formal definitions of (O -notation, Ω -notation and Θ -notation), 2.2 Understanding analysis framework, measuring an input's size and units for measuring running time), worst-case, best-case, and average-case efficiencies.
Goal 3	Concepts of Brute Force, sorting and Exhaustive Search
Learning Outcomes	3.1 Learning selection sort and bubble sort, insertion sort, mergesort, quicksort, and heapsort 3.2 Learning sequential search, Depth-first search and Breadth-first search. 3.3 Understanding binary search and insertion in a binary tree.
Goal 4	Concepts of Hashing
Learning Outcomes	4.1 Understanding Open Hashing (Separate Chaining) and Closed Hashing (Open Addressing) and B-Trees.
Goal 5	Concepts of Greedy Techniques
Learning Outcomes	5.1 Learning Prim's, Kruskal's, and Dijkstra's Algorithm 5.2 Learning Huffman Trees and Codes
Goal 6	Limitations of Algorithm Power
Learning Outcomes	6.1 Learning lower-bound arguments, trivial lower bounds, Decision trees for searching a sorted array and P , NP , and NP -Complete Problems, P and NP Problems.
Textbook	1. Anany Levitin, Introduction to the Design and Analysis of Algorithms, 3 rd ed., 2012, Pearson Education Inc.
Supplementary References	1. Michael T. Goodrich, Roberto Tamassia, Algorithm Design and Applications, John Wiley and sons, Inc., 2015. 2. M.T. Goodrich, R. Tamassia, M.H. Goldwasser, Data Structures and

Detailed Course Description - Course Plan Development and Updating Procedures/ Computer Information Systems Department	QF01/0408-3.0E
---	----------------

Algorithms in Java, 6 th ed. (International Student Ed.), 2014. 3. N. Dale, D. Joyce, C. Weems, Object-Oriented Data Structures Using Java, 3 rd ed., 2012.
--

Course Timeline				
Week	Number of Hours	Course Topics	Pages (Textbook)	Notes
01	1 1 1	Chapter 1: Introduction <ul style="list-style-type: none"> - What Is an Algorithm? - Fundamentals of Algorithmic Problem Solving - Understanding the Problem - Algorithm Design Techniques - Designing an Algorithm and Data Structures - Methods of Specifying an Algorithm - Analyzing an Algorithm - Coding an Algorithm - Important Problem Types - Fundamental Data Structures 	1-38	
02	1 1 1	Chapter 2: Fundamentals of the Analysis of Algorithm Efficiency <ul style="list-style-type: none"> - The Analysis Framework - Measuring an Input's Size - Units for Measuring Running Time - Worst-Case, Best-Case, and Average-Case Efficiencies - Recapitulation of the Analysis Framework - Asymptotic Notations and Basic Efficiency Classes - Informal Introduction (O-notation, Ω-notation and Θ-notation) 	41-58	
03	1 1 1	Chapter3: Brute Force and Exhaustive Search <ul style="list-style-type: none"> - Selection Sort - Bubble Sort - Sequential Search - Brute-Force String Matching - Closest-Pair and Convex-Hull Problems by Brute Force - Closest-Pair Problem - Convex-Hull Problem - Exhaustive Search - Traveling Salesman Problem - Knapsack Problem 	97-125	

Detailed Course Description - Course Plan Development and Updating Procedures/ Computer Information Systems Department	QF01/0408-3.0E
---	----------------

		<ul style="list-style-type: none"> - Assignment Problem - Depth-First Search - Breadth-First Search 		
04	1 1 1	<p>Chapter 4: Decrease-and-Conquer</p> <ul style="list-style-type: none"> - Insertion Sort - Topological Sorting - Algorithms for Generating Combinatorial Objects - Generating Permutations - Generating Subsets - Decrease-by-a-Constant-Factor Algorithms - Binary Search - Fake-Coin Problem - Variable-Size-Decrease Algorithms - Computing a Median and the Selection Problem - Interpolation Search - Searching and Insertion in a Binary Search Tree 	131-163	
05	1 1 1	<p>Chapter 5: Divide-and-Conquer</p> <ul style="list-style-type: none"> - Mergesort - Quicksort - Binary Tree Traversals and Related Properties - Multiplication of Large Integers and Strassen's Matrix Multiplication - Multiplication of Large Integers 	169-191	
06	1 1 1	<ul style="list-style-type: none"> - The Closest-Pair and Convex-Hull Problems by Divide-and-Conquer - The Closest-Pair Problem - Review of previous topics, solutions of problems. <p style="text-align: center;">First Exam</p>	192-195	
07	1 1 1	<p>Chapter 6: Transform-and-Conquer</p> <ul style="list-style-type: none"> - Presorting - Gaussian Elimination - LU Decomposition - Computing a Matrix Inverse - Balanced Search Trees - AVL Trees - 2-3 Trees - Heaps and Heapsort - Horner's Rule and Binary Exponentiation 	201-246	

Detailed Course Description - Course Plan Development and Updating Procedures/ Computer Information Systems Department	QF01/0408-3.0E
---	-----------------------

		<ul style="list-style-type: none"> - Problem Reduction - Computing the Least Common Multiple - Counting Paths in a Graph - Reduction of Optimization Problems - Linear Programming - Reduction to Graph Problems 		
08	1 1 1	Chapter 7: Space and Time Trade-Offs <ul style="list-style-type: none"> - Sorting by Counting - Input Enhancement in String Matching - Horspool's Algorithm - Boyer-Moore Algorithm 	253-267	
09	1 1 1	<ul style="list-style-type: none"> - Hashing - Open Hashing (Separate Chaining) - Closed Hashing (Open Addressing) - B-Trees 	269-274	
10	1 1 1	Chapter 8: Dynamic Programming <ul style="list-style-type: none"> - The Knapsack Problem and Memory Functions - Memory Functions - Optimal Binary Search Trees 	283-303	
11	1 1 1	<ul style="list-style-type: none"> - Warshall's and Floyd's Algorithms - Warshall's Algorithm - Floyd's algorithm for all-pairs shortest-paths problem <p style="text-align: center;">Second Exam</p>	304-311	
12	1 1 1	Chapter 9: Greedy Technique <ul style="list-style-type: none"> - Prim's Algorithm - Kruskal's Algorithm - Disjoint Subsets and Union-Find Algorithms - Dijkstra's Algorithm - Huffman Trees and Codes 	315-338	
13	1 1 1	Chapter 10: Iterative Improvement <ul style="list-style-type: none"> - The Simplex Method - Geometric Interpretation of Linear Programming - An Outline of the Simplex Method - Further Notes on the Simplex Method - The Maximum-Flow Problem - Maximum Matching in Bipartite Graphs 	345-380	
14	1 1 1	Chapter 11: Limitations of Algorithm Power <ul style="list-style-type: none"> - Lower-Bound Arguments - Trivial Lower Bounds - Information-Theoretic Arguments - Adversary Arguments 	387-397	

Detailed Course Description - Course Plan Development and Updating Procedures/ Computer Information Systems Department	QF01/0408-3.0E
---	-----------------------

		<ul style="list-style-type: none"> – Decision Trees for Sorting – Decision Trees for Searching a Sorted Array 		
15	1 1 1	<ul style="list-style-type: none"> – <i>P</i>, <i>NP</i>, and <i>NP</i>-Complete Problems – <i>P</i> and <i>NP</i> Problems – <i>NP</i>-Complete Problems – Review of previous topics, solutions of problems. 	401-409	
16	1 1 1	<ul style="list-style-type: none"> – Discussion of assignments 		
Final Exam				

Theoretical Course Evaluation Methods and Weight	Participation = 10% First Exam 20% Second Exam 20% Final Exam 50%	Practical (Clinical) Course Evaluation Methods	Semester Students' Work = 50% (Reports, Research, Quizzes, Etc.) Final Exam = 50%
---	--	---	---

Approved by Head of Department		Date of Approval	
---	--	-------------------------	--

Extra information (to be updated every semester by corresponding faculty member)

Name of Teacher		Office Number	
Phone Number (Extension)		Email	_____@zuj.edu.jo
Office Hours			