UNIVERSITY of NICOSIA

Course Syllabus

Course Code	Course Title	ECTS Credits
COMP-370	Algorithms	6
Prerequisites	**Department**	**Semester**
COMP-211 Data Structures	Computer Science	Fall
Type of Course	**Field**	**Language of Instruction**
Required	Computer Science	English
Level of Course	**Lecturer(s)**	**Year of Study**
1st Cycle	Prof. Athena Stassopoulou	3rd
Mode of Delivery	**Work Placement**	**Corequisites**
Face to Face	N/A	None

Course Objectives:

Presentation of data structures and algorithms that underpin much of today's computer programming. The study and analysis of various topics such as storage and execution time requirements, graph algorithms, minimum spanning trees, shortest paths, maximal matching, sorting, asymptotic analysis of recursive procedures, divide and conquer, local search algorithms.

Learning Outcomes:

After completion of the course students are expected to be able to:

1. Explain the big O and big omega notation and calculate the running times of programs
2. Implement lists, trees and sets and compare the various implementation approaches.
3. Implement the major sorting algorithms.
4. Design and implement an appropriate hashing function for an application.
5. Design and implement a collision-resolution algorithm for a hash table
6. Discuss the computational efficiency of the principal algorithms for sorting, searching, and hashing.
7. Solve problems using the fundamental graph algorithms, including depth-first and breadth-first search, single-source and all-pairs shortest paths, transitive closure, topological sort, and spanning tree algorithm.
8. Demonstrate the following capabilities: evaluate algorithms, select the appropriate

algorithm for solving a particular problem and justify the choice, and implement the algorithm in a programming context.

Course Content:

1. Design and analysis of algorithms, running time of a program, Big-Oh and Big-Omega Notation
2. Basic data types: lists (stacks and queues) and their implementations (arrays, linked lists)
3. Trees: Basic terminology, ordering of nodes, Implementation of Trees, binary trees, Huffman codes
4. Basic operations on sets and their implementation, Hashing (hash table, open and closed hashing, hash functions) , priority queues
5. Directed graphs: Single Source Shortest Path, All Pairs Shortest Path, Transitive Closure, Graph traversal, Topological Sorting, strong components
6. Undirected graphs: Minimum-Cost Spanning trees, traversals, graph matching
7. Sorting: bubblesort, insertion sort, selection sort, quicksort. Analysis of each sorting algorithm
8. Algorithm analysis techniques: Recursive programs, mergesort, solving recurrence equations.

Learning Activities and Teaching Methods:

Lectures, Practical Exercises and Assignments

Assessment Methods:

Mid-term exam, Projects, Assignments, Final Exam.

Required Textbooks / Readings:

Title	Author(s)	Publisher	Year	ISBN
Algorithms	S. Dasgupta, C. Papadimitriou and U. Vazirani	McGraw-Hill	2008	978-0-07-352340-8

UNIC

Recommended Textbooks / Readings:

Title	Author(s)	Publisher	Year	ISBN
Data Structures and Algorithms	A. V. Aho, J. E. Hopcroft and J. D. Ullman	Addison-Wesley	1983	978-0201-0000238
Algorithms (4th ed.)	R. Sedgewick and K. Wayne	Addison-Wesley Professional	2011	032157351X