East West University
Department of Computer Science and Engineering
Course Outline of CSE246

Course: CSE246 Algorithms (Sections: All)

Credits and Teaching Scheme

	Theory	Laboratory	Total
Credits	3	1.5	4.5
Contact Hours	3 Hours/Week for 13 Weeks + Final Exam in the 14th week	3 Hours/Week for 13 weeks	6 Hours/Week for 13 Weeks + Final Exam in the 14th week

Prerequisite

CSE103 Structured Programming
CSE207 Data Structured

Instructor Information

Dr. Tania Sultana
Assistant Professor, Dept. of CSE
East West University

Course Objective

This course introduces students to the general tools and techniques for analyzing and designing computer algorithms. Initially necessary mathematical preliminaries required for analyzing and designing computer algorithms are taught. Then this course familiarizes students with several algorithmic approaches and corresponding problems. This course will work as a backbone to understanding different core courses of computer science and will be needed as prerequisite knowledge for future courses such as Artificial Intelligence, Computer Networks, and Compiler Design.

Knowledge Profile

K4: Forefront engineering specialist knowledge for practice

Learning Domains

Cognitive – C3: Applying, C4: Analyzing
Psychomotor - P2: Manipulation, P3: Precision
Affective - A2: Responding

Program Outcomes (POs)

PO1: Engineering Knowledge
PO2: Problem Analysis

Complex Engineering Problem

EP1: Depth of knowledge required.
EP2: Range of conflicting requirements.
EP3: Depth of analysis required.

Course Outcomes (COs) with Mappings

After completion of this course students will be able to:

CO	CO Description	PO	Learning Domains	Knowledge Profile	Complex Engineering Problem Solving/ Engineering Activities
CO1	**Model** different real-life problems using graph and **apply** graph related algorithms to solve them.	PO1	C3	K4	
CO2	**Apply** the basic concepts of number theory, pattern matching for developing effective problem solutions.	PO1	C3	K4	
CO3	**Choose** and **justify** Advanced algorithm design techniques for solving complex problems.	PO2	C4	K4	EP1, EP2, EP3
CO4	**Analyze** the complexity of different algorithms and **choose** the suitable approach for solving complex problems.	PO2	C4	K4	

Course Topics, Teaching-Learning Method, and Assessment Scheme

Course Topic	**Teaching-Learning Method**	**CO**	**Mark of Cognitive Learning Levels**			**Exam (Mark)**
			C2	C3	C4	
Introduction to algorithms, complexity analysis, asymptotic notations, typical running time functions, classifying. functions by their asymptotic growth rates, etc.		CO4	3			Midterm (30)
Divide and conquer algorithms: Binary search, Closest pair of points, Counting inversion, merge sort, quick sort etc.		CO3	2	4		
Greedy algorithms: Coin changing, fractional Knapsack, Huffman codes, Optimal codes, Activity selection. Suitability of all these algorithms in greedy approach.			2	4		
Recurrence relation. Iteration, Substitution, Recursion tree and Master methods		CO4			3	
Euclid's algorithm for GCD, Extended Euclid's algorithm and Number theoretic algorithms: Sieve Method. Pattern matching and String-matching algorithms		CO2	3	3		

(Rabin-Karp Algorithm, KMP Algorithm).						
Introduction to dynamic programming: Fibonacci series, 0-1 knapsack, coin changes and related problems, Longest Common Subsequence (LCS) and related problem		CO3		6		
Dynamic programming (Continue): Longest Increasing subsequence (LIS), Rock climbing, matrix chain multiplication, and miscellaneous problem solving.		CO3		6		Theory final (30)
Graphs, graph-based algorithm - breadth-first search (BFS), depth-first search (DFS), Edge identification. Modification of DFS to find the topological sort, strongly connected component, articulation points, bridges, and Bi-connected components.		CO1	2	6		
Minimal spanning tree: basic terminology, applications and algorithms Single source shortest path algorithms: Dijkstra's algorithm, Bellman-Ford algorithm, shortest path in DAG. Floyd-Warshall algorithm, Transitive closure. Algorithm strategy, structure and problem types.		CO1	2	6		
Network Flow, Max Flow, Min-Cut, Residual Network, Augmenting paths, Ford-Fulkerson and Edmonds-Karp algorithms.		CO1	2	3		
P and NP classes, algorithm completeness, discussion on other complex techniques of		CO4		3		

algorithm design And analysis.						

Laboratory Experiments and Assessment Scheme

Experiment	**Teaching-Learning Method**	**CO**	**Mark of Cognitive Learning Levels**		**Mark of Psychomotor Learning Levels**		**Mark of Affective Learning Levels**	**CO Mark**
			C3	**C4**	**P2**	**P3**	**A2**	
Implementation of Divide and Conquer Algorithms	Do	CO3						
Implementation of Greedy- Knapsack (fractional) and Huffman codes	Do	CO3						
Implementation of String Matching	Do	CO2						
Implementation of Sieve	Do	CO2						
Implementation of DP- Knapsack, LCS, LIS, Coin change, Matrix chain multiplication and other DP related problem	Do	CO3						
Implementation of breadth-first search (BFS)	Do	CO1						
Implementation of depth-first search (DFS)	Do	CO1						

Implementation of Topological sort and find Strongly connected component	Do	CO1						
Implementation of Dijkstra's and Modified Dijkstra's algorithms	Do	CO1						
Implementation of Floyd-Warshall algorithm and Transitive closure	Do	CO1						
Implementation of Max Flow	Do	CO1						
Lab Performance	Individual lab evaluation							10
Lab Exam	Individual lab exam							10
Total								

Mini Projects

Mini Project	**Teaching-Learning Method**	**CO**	**Mark of Cognitive Learning Levels**		**Mark of Psychomotor Learning Levels**		**Mark of Affective Learning Levels**	**CO Mark**
			C3	**C4**	**P2**	**P3**	**A2**	
Mini Lab Project including Report and Presentation	Group based moderately complex project with report writing, and oral/poster presentation	CO3						10

Overall Assessment Scheme

Assessment Area	CO1	CO2	CO3	CO4	others	Assessment Area Mark
Class Test	2.5	2.5	3	2		10
Midterm		6	18	6		30
Final Exam (Theory)	21		6	3		30
Lab final	6		4			10
Lab performance	3	3	4			10
Mini Project			10			10
Total Mark	32.5	11.5	45	11		100

Teaching Materials/Equipment

Text Book	1. Introduction to Algorithm (3 rd edition) by Thomas H. Cormen, Charles E. Leiserson, Ronald Rivest, Clifford Stein 2. Algorithms by Robert Sedgewick
Teaching Materials	Lecture notes*, Lab exercise/ assignments/notes*, Reference books, Computer and Software

- Slides/Course Materials will be provided/discussed during classes

Mini Projects:
Mini Project description will be provided in time.

Grading System

Marks (%)	Letter Grade	Grade Point	Marks (%)
80-100	A+	50-54	C+
75-79	A-	45-49	C
70-74	B+	40-44	D
65-69	B	0-39	F
60-64	B		
55-59	B-		

Exam Dates

Can be found on the website.

Academic Code of Conduct

Academic Integrity:
Any form of cheating, plagiarism, personification, falsification of a document as well as any other form of dishonest behavior related to obtaining academic gain or the avoidance of evaluative exercises committed by a student is an academic offence under the Academic Code of Conduct and **may lead to severe penalties as decided by the Disciplinary Committee of the university.**

Special Instructions:

- Students are expected to attend all classes and examinations. A student MUST have at least 80% class attendance to sit for the final exam.
- Students will not be allowed to enter into the classroom after 20 minutes of the starting time.
- For plagiarism, the grade will automatically become zero for that exam/assignment.
- Normally there will be **NO make-up exam**. However, in case of **severe illness, death of any family member, any family emergency, or any humanitarian ground**, if a student miss any exam, the student MUST get approval of makeup exam by written application to the Chairperson through the Course Instructor **within 48hours**of the exam time. Proper supporting documents in favor of the reason of missing the exam have to be presented with the application.
- For **final exam**, there will be NO makeup exam. However, in case of **severe illness, death of any family member, any family emergency, or any humanitarian ground**, if a student miss the final exam, the student MUST get approval of **Incomplete Grade** by written application

to the Chairperson through the Course Instructor **within 48 hours** of the final exam time. Proper supporting documents in favor of the reason of missing the final exam have to be presented with the application. **It is the responsibility of the student to arrange an Incomplete Exam within the deadline mentioned in the Academic Calendar in consultation with the Course Instructor**.

- All mobile phones MUST be turned to silent mode during class and exam period.
- There is **zero tolerance for cheating** in exam. Students caught with cheat sheets in their possession, whether used or not; writing on the palm of hand, back of calculators, chairs or nearby walls; copying from cheat sheets or other cheat sources; copying from other examinee, etc. would be treated as cheating in the exam hall. The only penalty for cheating is **expulsion for several semesters as decided by the Disciplinary Committee of the university**.