University of Asia Pacific (UAP)

Department of Computer Science and Engineering (CSE)

BSc in CSE Program

Course Outline – Data Structures and Algorithms II Lab

Part A – Introduction

1. **Course No. / Course Code:** CSE 208
2. **Course Title:** Data Structures and Algorithms II Lab
3. **Course Type:** Core Course
4. **Level/Term and Section:** 2nd year 2nd semester
5. **Academic Session:** Spring 25
6. **Course Instructor: Fabliha Haque (**Lecturer**), Sadia Jahangir Safa (**Lecturer**),**
7. **Pre-requisite (If any):** Nil
8. **Credit Value:** 1.5
9. **Contact Hours:** 3.0
10. **Total Marks:** 100
11. **Course Objectives and Course Summary:**
    Data structures and algorithms are essential for the students to be able to design, develop efficient algorithms using different data structures. This course is designed to help the students learn the skills to design and implement efficient algorithms using appropriate data structures. This is a required course and a prerequisite of Operating System, and Artificial Intelligence and Expert System in the CSE program.
    The prime objective of this course is to provide students with an in-depth knowledge of applying data structures and algorithms to solve different types of problems.
12. **Course Learning Outcomes: at the end of the Course, the Student will be able to –**

CLO 1	**Explain** terms related to important data structures, algorithm analysis, design techniques, and basic algorithms.

CLO 2	**Apply** techniques and appropriate data structures to design and implement algorithms to solve a practical problem.
CLO 3	**Analyze** the performance or resource requirements of various algorithms.
CLO 4	**Apply** algorithmic techniques and data structures through interactive calculation and visualization.

13. Mapping / Alignment of CLOs with Program Learning Outcomes (PLO) (Optional):

CLO No.	Corresponding PLOs (Appendix-1)	Bloom's taxonomy domain/level (Appendix-2)	Delivery methods and activities	Assessment Tools
CLO1	1	1/Remember	Coding, Multimedia	Quiz, Assignment
CLO2	3	1/Apply	Coding, Multimedia, Problem solving session	Quiz, Viva, Assignment, Lab Test
CLO3	2	1/Analyze	Coding, Multimedia	Quiz, Assignment, Lab Test
CLO4	3	1/ Apply	Coding, Multimedia, Illustrations	Quiz, Assignment

Part B – Content of the Course

14. Course Content:

Introduction: The role of algorithms in computing. **Complexity analysis**: Growth of function, asymptotic notations, orders, designing worst case and average-case. **Recurrence relations**: Substitution method, iteration method, master method. **Divide and Conquer**: Basic idea, control structure properties of D&C, **Applications of D&C:** Binary Search, Ternary Search, Merge Sort, Quick Sort. **Dynamic Programming**: Elements of Dynamic Programming, Comparison with D&C. **Application of Dynamic programming in**: Longest Common Subsequence, 0/1 Knapsack problem. **Greedy Method**: Elements of greedy method, basic control structure, **Application of Greedy method in**: Coin Change, Fractional Knapsack Problem, Job sequencing with deadline. **Graph related algorithms**: Breadth First search, Depth First search, Dijkstra's shortest path algorithm, The Bellman-Ford algorithm, The Floyd-War shall algorithm. **Backtracking**: Basic idea behind backtracking, control structure. **Application of backtracking in**: graph coloring problem, n -queens' problems. **Branch and Bound**: Basic idea and control structure of Branch and Bound. FIFO branch and bound, LC Branch and Bound, the l5-puzzle problem. **String Matching**: Naïve string-matching algorithm, the Rabin-Karp algorithm. **Approximation Algorithms**: Introduction, the vertex-cover problem, the traveling salesman problem, the subset-sum problem. **Completeness**: Polynomial time, polynomial time verification, NP-completeness and reducibility, NP completeness proofs, NP complete problems.

15. Alignment of topics of the courses with CLOs:

SL. No	Topics / Content	Course Learning Outcome (CLO)

1	Introduction to algorithms	CLO1
2	Algorithm Implementation: Divide and Conquer, Dynamic Programming, Greedy algorithms, Graph Algorithms, Backtracking, String Matching, Approximation Algorithms	CLO2
3	Complexity Analysis, Analysis of Algorithms: Divide and Conquer, Dynamic Programming, Greedy algorithms, Graph Algorithms, Backtracking, String Matching, Approximation Algorithms	CLO3
4	Numerical Calculation and visualization: Divide and Conquer, Dynamic Programming, Greedy algorithms, Graph Algorithms, Backtracking, String Matching, Approximation Algorithms	CLO4

16. Class Schedule/Lesson Plan/Weekly plan:

Topics	**Specific Outcome(s)**	**Time Frame**	**Suggested Activities**	**Teaching Strategy(s)**	**Alignment with CLO**
Introduction, Complexity Analysis: Worst-Case and Best-Case Analysis, Asymptotic notation.	- To analyze the time complexity of algorithms	Week 1	Visualization of different functions, numerical analysis of functions	Practice session	CLO1, CLO3
The Divide-and-Conquer (DC) Approach: Binary Search, Ternary Search, Merge Sort, Quick Sort, Analyzing DC Algorithms, Recurrence Relations, Master Method.	- To apply DC algorithms for real life problems - To analyze time complexity of DC using different methods	Week 2 – Week 4	Implementing DC algorithms and complexity analysis	Practice session, Coding	CLO2, CLO3, CLO4
Dynamic Programming: Longest Common Subsequence, Coin changing problem,	- To apply DP and memoization	Week 5 – Week 6	Implementation and Time Complexity	Practice session, Coding	CLO2, CLO3, CLO4

0/1 Knapsack Problem	- To identify overlapping subproblems		analysis DP algorithms		
Greedy algorithms: Coin change, fractional Knapsack, Job Sequencing	- To apply greedy algorithms - To minimize time complexity using Greedy approach	Week 7	Code implementation of problems which can be solved using greedy method	Lecture, multimedia, Group discussion	CLO2, CLO4
Graph Related Algorithms: DFS, BFS, Single Source Shortest Path, Dijkstra's Algorithm, MST	- To understand graph representation and traversal	Week 8 – Week 9	Solve graph related problems	Practice session, Coding, Homework	CLO2, CLO4
Backtracking: Graph Coloring, N-Queen problem	- To apply backtracking idea	Week 10	Implement Backtracking for N-Queen problem	Practice session, Coding, Homework	CLO2, CLO4
Branch and Bound: FIFO branch and Bound, LC Branch and Bound, the l5-puzzle problem.	- To understand branch and bound concept	Week 11	Solve 15 puzzle by hand and note the moves then implement the algorithm	Practice session, Coding	CLO2, CLO4
String Algorithms: The Naive String-Matching Algorithm, The Rabin-Karp Algorithm	- To develop efficient string-matching algorithms	Week 12 – Week 13	Make a hash function and use it in implementing Rabin Karp algorithm	Practice session, Coding	CLO2, CLO3, CLO4
Approximation Algorithms: the vertex-cover problem, the traveling-salesman problem, the subset-sum problem	- To develop approximation algorithms - To find the approximation ratio	Week 14	Implement Approximation algorithms	Practice session, Coding	CLO2, CLO3, CLO4

FINAL LAB TEST					

17. Teaching-Learning Strategies:

18. Assessment Techniques of each topic of the course:

Assessment Type	% weight	CO1	CO2	CO3	CO4
Final Assessment (Lab Test, Quiz, Viva)	40%	5	20	5	10
Continuous Assessment (Lab Test, Assignment, Problem-solving, Quiz)	60%	5	40	5	10
Total	**100%**	**10**	**60**	**10**	**20**

Part C – Assessment and Evaluation

19. Assessment Strategy

Lab Tests: Altogether 4 lab tests will be taken during the semester, 4 tests will be taken before midterm and 2 class tests will be taken before final term. A mid-assessment will be conducted in the 7th week, and a final assessment will take place in the 14th week. No makeup lab tests will be taken. Students are strongly recommended not to miss any lab tests.

Assignment: The students will have to form a group of maximum 3 members. The topic or case studies will be given as assignments in groups during the class which they have to prepare at home and will submit on or before the due date. No late submission of assignments will be accepted. Students will have to do the presentation on the given topic as assignment

CIE- Continuous Internal Evaluation (40 Marks)

Bloom's Category	Lab Tests (40)	Assignments (20)
Remember		
Understand		5
Apply	25	5

Analyze	5	5
Evaluate		
Create	10	5

SMEB- Semester Mid & End Examination (60 Marks)

Bloom's Category	Lab Final (40)
Remember	
Understand	5
Apply	10
Analyze	5
Evaluate	
Create	10

20. Evaluation Policy

Grades will be calculated as per the university grading structure and individual student will be evaluated based on the following criteria with respective weights.

1. Lab Assessment 60%
2. Lab Final Examination 40%

UAP Grading Policy

Numeric Grade	Letter Grade	Grade Point
80% and above	A+	4.00
75% to less than 80%	A	3.75
70% to less than 75%	A-	3.50
65% to less than 70%	B+	3.25
60% to less than 65%	B	3.00
55% to less than 60%	B-	2.75
50% to less than 55%	C+	2.50
45% to less than 50%	C	2.25
40% to less than 45%	D	2.00

Less than 40%	F	0.00

Part D – Learning Resources

21. Text Book

1. Ellis Horowitz, S. Rajasekaran, Sartaj Sahni: Fundamental of Computer Algorithms, Second Edition, 2008, Universities Press.

Reference Books & Materials

2. T.H. Cormen, C.E.Leiserson, R. L. Rivest, C. Stein: Introduction to Algorithms, Third Edition, 2009, PHI Learning Pvt. Ltd