Collage of Computing and Information Technology
Department of Computer Science

COURSE OUTLINE – 2023

1	**Course Title**	Design and Analysis of Algorithms
2	**Course Code**	CS-401
3	**Credit Hours**	3
4	**Semester**	Semester 7
5	**Lecturer**	Suleiman Gargaare (MSc in CS, MA in PM)
6	**Lecturer Contact**	suleiman.gargaare@uoh.edu.so
7	**Contact Hours**	3 hours per week
8	**Schedule**	**4C**: Sat 04:00-07:30PM, **4D**: Sun 04:00-07:30PM

9 Course Introduction

This course is designed to enable students to understand the design, complexity and efficiency issues related to different algorithms. It is an introductory course to the design and analysis of algorithms. It covers mainly asymptotic analysis of time complexity, Proofs of correctness, Algorithms and advanced data structures for searching and sorting lists, and graph algorithms. Some advance topics like dynamic programming and greedy algorithms with their applications will also be introduced.

10 Learning Objectives

- The student will develop knowledge of
  - intermediate level algorithms and data structures
  - techniques for the analysis of their complexity and correctness
  - some of the fundamental limitations on what can and cannot be computed efficiently.
- The student will be expected to learn
  - models for asymptotic analysis of algorithms;
  - techniques for searching and sorting lists including its analysis
  - various sorts such as insertion, selection, merge, quick, count heap and linear sorts;
  - lower/upper bounds on the complexity of sorting;
  - Some problem solving techniques having multiple solution i.e. dynamic programming and greedy algorithms

11 Course Contents

Detailed study of the basic notions of the design of algorithms and the underlying data structures. Several measures of complexity are introduced. Emphasis on the structure, complexity, and efficiency of algorithms.

12 Lecture Schedule

Weeks	Topic of Lecture	Reading Assignment
Week 1 & 2	***Introduction to Algorithms:*** Definition of Algorithm and its role in computing, its implementation domain, data structure & techniques, Introductionto Algorithm analysis and its importance, RAM model, methods for analyzing algorithms, algorithm efficiency, complexity and order of growth, and some other case studies, benchmarks for analysis, complexity classes, Big – O Notation.	Yes

Week 3	***Growth of Functions:*** Asymptotic notations, ***O,*** Θ and Ωnotations, asymptotic notation in equations and inequalities, ***o***–notation, **ω**-notation, comparison of functions, standardnotations & common function. Association of asymptotic notationto complexity classes	None
Week 4 & 5	***Designing and analysis of searching algorithms:*** Searching Algorithms, structure of data, why searching is used? Different searching techniques, analysis of DFS, BFS, GBFS and A* Search,best, worst & average case analysis; complexity, optimality and the correctness of Searching Algorithms, ***Quiz 01***	Yes
Week 6 & 7	***Designing and analysis of sorting algorithms:*** Sorting and order statistics, structure of data, why sorting is used? Different sorting techniques, analysis of insertion, selection and bubble sort,best, worst & average case analysis; loop invariants and the correctness, incremental approach, divide and conquer approach,its use in merge and quick sort; description and analysis of mergeand quick sort, performance of merge and quick sort, worst, average & best cases, analysis of merge and quick sort. ***Assignment 01***	Yes
Week 8	***Student Week*** **| Discussions and Presentations**	
Week 9	***Mid Exam***	
Week 10 & 11	***Divide and Conquer:*** Elements of the Divide and Conquer strategy, Recurrence Relation for Decreasing Function, Master Theorem for Decreasing Function, Frequency Count Method, Recurrence for Root Function, Recurrence Relation for Dividing Function, Master Theorem for Dividing Function	Yes
Week 12	***Greedy Algorithms:*** Elements of the greedy strategy, greedy choice property, optimal substructure, greedy versus dynamic programming. Activity Selection Problem, Making Change, Huffman Codes, prefix code, Construction of Huffman code, andCorrectness of Huffman's algorithm. ***Quiz 02***	None
Week 13	***Dynamic Programming:*** Assembly-line scheduling, fastest way through the factory, recursive solution, computation of the fastesttime, construction of fastest way, Matrix–chain Multiplication, optimal parenthesization, recursive solution, optimal cost calculation, construction of optimal solution. 0-1 Knapsack Problem, recursive solution, optimal cost calculation, construction of optimal solution.	None

Week 14	**Graphs:** Shortest path calculation, Minimum Spanning Trees, Prims Algorithm, Kruskal Algorithm, Djikstras Algorithm ***Assignment 02***	Yes
Week 15	**Revision of the course**	

13 Course Assessment

The assessment of this course shall have following breakdown structure

Quizzes	10%
Assignments	15%
Mid Exam	15%
Final Exam	60%

The minimum pass marks for each course shall be 50%.
Students obtaining less than 50% marks in any course shall be deemed to have failed in that course.

14 Attendance Policy

Every student **must attend 80%** of the lectures/discussions delivered in this course and 80% of the practical/laboratory work prescribed for the respective courses. The students falling short of required percentage of attendance of lectures/discussion/practical/lab work, etc., shall not be allowed to appear in the final examination of this course and shall be treated as having failed this course.

15	**Text Book**	**I.** T. H. Cormen, C. E. Leiserson, and R. L. Rivest (2009), *Introduction to Algorithms*, (3rd Ed.) MIT Press, McGraw-Hill, New York.
16	**Reference Books**	**I.** Algorithms, Robert Sedgewick, Edition 2nd, **II.** Compared to What? An Introduction to the Analysis of Algorithms, by G.J.E Rawlins **III.** The Design and Analysis of Computer Algorithms, byAlfred V. Aho