Tunisian Republic
Private Higher School of Engineers of Gafsa
Private Higher Education Institution,
State-approved under N° 05-2013

Course Title:	**Algorithm design and analysis**
Course Code:	CSE321
Program:	Computer science Engineering
Department:	Computer Engineering
Course coordinator:	Dr. Dhekra CHERMITI
Institution:	Private Higher School of Engineers of Gafsa (ESIP)

A. Course Identification

1. Credit hours: **3 (2-1-0)**

2. Course type

a. College □ Department ■ Others □

b. Fundamental ■ Transversal □ Optional □

3. Level/year at which this course is offered: 3/3

4. Pre-requisites for this course : CSE131, CSE132, UML, Object Oriented Programming

1. Mode of Instruction (mark all that apply)

No	Mode of Instruction	Contact Hours	Self-study	Total workload
1	**Traditional classroom**	…..	35	80
2	**Blended**	45		
3	**E-learning**	……		
4	**Distance learning**	……		
5	**Other ()**	……		

2. Contact Hours (based on academic semester)

No	Activity	Contact Hours
1	**Lecture**	22.5
2	**Laboratory/Studio**	-
3	**Tutorial**	22.5
4	**Others** (specify)	-
	Total	45

B. Course Objectives and Learning Outcomes

Course Description

This course covers fundamental techniques for designing and analyzing algorithms. Students will learn about algorithm complexity, problem classifications, and computational models. Key topics include recursive and iterative algorithms, polynomial transformations, and major algorithmic paradigms such as divide and conquer, greedy methods, and dynamic programming. The course also introduces advanced data structures like AVL and balanced trees, helping students evaluate algorithm efficiency and solve complex computational problems.

Course Main Objective

Course Aims:

✓ Develop a strong foundation in algorithm complexity.
✓ Understand and implement iterative and recursive algorithms.
✓ Enhance problem-solving skills through algorithm design and analysis.
✓ Apply algorithmic principles to identify, model, and solve complex computational problems.

1. Course Learning Outcomes

CLOs		Aligned PLOs
1	**Knowledge and Understanding**	
1.1	- Develop a foundational understanding of algorithm complexity and its impact on computational efficiency.	PLO.K1
1.2	- Understand and apply iterative and recursive algorithms for problem-solving and computational analysis.	PLO.K2
2	**Skills**	
2.1	- Utilize algorithm design and analysis techniques to develop efficient solutions for complex problems.	PLO.S1
2.5	- Apply algorithmic principles to identify, model, and evaluate computational problems in real-world scenarios.	PLO. S5

C. Course Content

No	List of Topics	Contact Hours
1	Complexity of algorithms	2
2	Iterative algorithms Recursive algorithms	2.5
3	Complexity of problems	2.5
4	Turing machine P, ZPP, BPP classes	2.5
5	Polynomial transformation	2
6	Programming paradigms	2
7	Divide and conquer approach	2
8	Greedy method	2
9	Dynamic programming	2.5

	Balanced trees	
10	AVL trees / two-tone trees	2.5
Total		22.5

C .1 Tutorial Content

No	List of Topics	Contact Hours
1	Complexity of algorithms, Iterative and Recursive algorithms	4.5
2	Turing machine	4.5
3	Divide and conquer approach	4.5
4	Polynomial transformation	4.5
5	Balanced trees	4.5
Total		22.5

D. Teaching and Assessment

1. Alignment of Course Learning Outcomes with Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
1.0	**Knowledge and Understanding**		
PLO.K1	- Develop a foundational understanding of algorithm complexity and its impact on computational efficiency.	- Lecturing	- Assignments, Quizzes , Exams,
PLO.K2	- Understand and apply iterative and recursive algorithms for problem-solving and computational analysis.		
2.0	**Skills**		
PLO.S1	- Utilize algorithm design and analysis techniques to develop efficient solutions for complex problems.	- Lecturing	- Assignments, Quizzes , Exams,
PLO.S5	- Apply algorithmic principles to identify, model, and evaluate computational problems in real-world scenarios.		

2. Assessment Tasks for Students

#	Assessment task*	Week Due	Percentage of Total Assessment Score
1	Practical Work (written or oral)	Weekly	00%
2	Quizzes, Homework assignments	Random	00%
3	First mid Term	8	35%
4	Final Exam	16	65%

E. Student Academic Counselling and Support

Arrangements for availability of faculty and teaching staff for individual student consultations and academic advice:
- Office hours - Blackboard interface

F. Learning Resources and Facilities

1. Learning Resources

Required Textbooks	1. Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. *Introduction to Algorithms*, MIT Press. 2. Kleinberg, J., & Tardos, É. – *Algorithm Design*, Pearson. 3. Wegener, I. – *Complexity Theory: Exploring the Limits.* 4. Möhring, R. H., & Raman, R. – *Efficient Algorithms*, Springer, 2005
Essential References Materials	- NA
Electronic Materials	1. Online lecture slides and course notes (provided via LMS). 2. Algorithm visualization tools (e.g., VisuAlgo, AlgoLab). 3. Research papers and online tutorials (ACM Digital Library, IEEE Xplore)
Other Learning Materials	- NA

2. Facilities Required

Item	**Resources**
Accommodation	**Classroom board** **Computer lab with the necessary software** **Internet access**
Technology Resources	**Data projector**

G. Course Quality Evaluation

Evaluation Areas/Issues	**Evaluators**	**Evaluation Methods**
Effectiveness of teaching and assessment.	Students, course coordinator, Alumni, Employers	Direct/Indirect
Extent of achievement of course learning outcomes.	Faculty, Program Leaders, quality department	Direct
Quality of Learning resources	Faculty, Program Leaders,	Direct, Indirect
Teaching and learning quality and effectiveness.	Students, Faculty Program Leaders,	Direct, Indirect

H. Specification Approval Data

Council / Committee	Computer Engineering Council
Date	11/09/2023