



National University of Engineering (UNI)
School of Computer Science
Syllabus 2027-I

1. COURSE

CS1D1. Discrete Structures (Mandatory)

2. GENERAL INFORMATION

2.1 Course	: CS1D1. Discrete Structures
2.2 Semester	: 2 nd Semester.
2.3 Credits	: 4
2.4 Hours	: 2 HT; 4 HP;
2.5 Duration of the period	: 16 weeks
2.6 Type of course	: Mandatory
2.7 Learning modality	: Face to face
2.8 Prerequisites	: None

3. PROFESSORS

Meetings after coordination with the professor

4. INTRODUCTION TO THE COURSE

Discrete structures provide the theoretical foundation for computer science. This course introduces the essential mathematical tools used to model computational problems, including logic, set theory, and functions. Understanding these concepts is critical for developing rigorous reasoning skills and for the subsequent study of algorithms, databases, and formal languages.

5. GOALS

- Understand the principles of mathematical logic and proof techniques.
- Perform operations with sets, functions, and relations.
- Apply basic counting techniques and recurrence relations to problems.
- Develop the ability to think abstractly and solve problems formally.

6. COMPETENCES

AG-C07) Computing Knowledge: Applies knowledge of mathematics, science, and computing. (Usage)

AG-C12) Uses systemic approaches to select, develop, apply, integrate, and manage secure computing technologies to achieve user objectives. (Usage)

7. TOPICS

Unit 1: Discrete Mathematics (12 hours)	
Competences Expected: AG-C07,AG-C12	
Topics	Learning Outcomes
• Logic: propositional and predicate logic, quantifiers, proofs • Sets: operations, cardinality, power sets, Cartesian products • Relations: properties, equivalence relations, partial orders • Functions: injective, surjective, bijective, composition, inverse • Advanced logic: proof systems, completeness, decidability • Advanced set theory: infinite sets, Cantor's diagonal argument	• Explain propositional and predicate logic [Explain] • Construct simple logical proofs [Apply] • Define basic set operations and relations [Explain] • Define properties of relations [Explain] • Define types of functions [Explain]
Readings : [Rosen2019], [Lehman2018]	

Unit 2: Discrete Mathematics (12 hours)	
Competences Expected: AG-C07,AG-C12	
Topics	Learning Outcomes
• Logic: propositional and predicate logic, quantifiers, proofs • Advanced logic: proof systems, completeness, decidability • Advanced set theory: infinite sets, Cantor's diagonal argument • Advanced relations: closures, lattices • Advanced functions: recursive functions, computability	• Construct simple logical proofs [Apply] • Explain concepts of infinite sets and cardinality [Explain] • Apply closure operations to relations [Apply] • Determine if a relation is an equivalence relation [Apply]
Readings : [Rosen2019], [Velleman2019]	

Unit 3: Discrete Mathematics (12 hours)	
Competences Expected: AG-C07,AG-C12	
Topics	Learning Outcomes
• Relations: properties, equivalence relations, partial orders • Functions: injective, surjective, bijective, composition, inverse • Graphs: definitions, paths, cycles, trees, connectivity • Advanced graph theory: planarity, coloring, networks	• Determine if a relation is an equivalence relation [Apply] • Compose functions and find inverses [Apply] • Define basic graph concepts [Explain] • Solve basic graph problems [Apply] • Analyze advanced graph properties [Evaluate]
Readings : [Rosen2019], [Scheinerman2012]	

Unit 4: Discrete Mathematics (12 hours)	
Competences Expected: AG-C07,AG-C12	
Topics	Learning Outcomes
- Counting: permutations, combinations, binomial theorem - Advanced counting: generating functions, inclusion-exclusion - Recurrence relations: solving simple recurrences - Advanced recurrence: generating functions, asymptotic behavior	- Explain permutations and combinations [Explain] - Apply counting techniques to solve problems [Apply] - Define recurrence relations [Explain] - Solve simple recurrence relations [Apply] - Develop generating functions for counting problems [Create]
Readings : [Rosen2019], [Grimaldi2003]	

8. WORKPLAN

8.1 Methodology

Individual and team participation is encouraged to present their ideas, motivating them with additional points in the different stages of the course evaluation.

8.2 Theory Sessions

The theory sessions are held in master classes with activities including active learning and roleplay to allow students to internalize the concepts.

8.3 Practical Sessions

The practical sessions are held in class where a series of exercises and/or practical concepts are developed through problem solving, problem solving, specific exercises and/or in application contexts.

9. EVALUATION SYSTEM

Theory Sessions:

The theory sessions are held in master classes with activities including active learning and roleplay to allow students to internalize the concepts.

Lab Sessions:

In order to verify their competences, several activities including active learning and roleplay will be developed during lab sessions.

Oral Presentations:

Individual and team participation is encouraged to present their ideas, motivating them with additional points in the different stages of the course evaluation.

Reading:

Throughout the course different readings are provided, which are evaluated. The average of the notes in the readings is considered as the mark of a qualified practice. The use of our virtual campus allows each student to access the course information, and interact outside the classroom with the teacher and with the other students.

The final grade NF depends on the following grades:

- PE : Partial Exam.
- FE : Final Exam.
- P : Permanent Evaluation (Average of practical work and graded assignments).

The final grade NF is calculated as follows:

$$NF = \frac{PE + FE + P}{3}$$

10. BASIC BIBLIOGRAPHY