National University of Engineering (UNI)
School of Computer Science
Syllabus 2026-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 Prerrequisites	:	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: Matemáticas Discretas (12 hours)	
Competences Expected: AG-C07,AG-C12	
Topics	**Learning Outcomes**
• Lógica: lógica proposicional y de predicados, cuantificadores, demostraciones • Conjuntos: operaciones, cardinalidad, conjuntos potencia, productos cartesianos • Relaciones: propiedades, relaciones de equivalencia, órdenes parciales • Funciones: inyectivas, sobreyectivas, biyectivas, composición, inversa • Lógica avanzada: sistemas de demostración, completitud, decidibilidad • Teoría avanzada de conjuntos: conjuntos infinitos, argumento diagonal de Cantor	• Explicar lógica proposicional y de predicados [Explicar] • Construir demostraciones lógicas simples [Aplicar] • Definir operaciones y relaciones básicas de conjuntos [Explicar] • Definir propiedades de las relaciones [Explicar] • Definir tipos de funciones [Explicar]
Readings : [Ros19], [LLM18]	

Unit 2: Matemáticas Discretas (12 hours)	
Competences Expected: AG-C07,AG-C12	
Topics	**Learning Outcomes**
• Lógica: lógica proposicional y de predicados, cuantificadores, demostraciones • Lógica avanzada: sistemas de demostración, completitud, decidibilidad • Teoría avanzada de conjuntos: conjuntos infinitos, argumento diagonal de Cantor • Relaciones avanzadas: clausuras, retículos • Funciones avanzadas: funciones recursivas, computabilidad	• Construir demostraciones lógicas simples [Aplicar] • Explicar conceptos de conjuntos infinitos y cardinalidad [Explicar] • Aplicar operaciones de clausura a relaciones [Aplicar] • Determinar si una relación es una relación de equivalencia [Aplicar]
Readings : [Ros19], [Vel19]	

Unit 3: Matemáticas Discretas (12 hours)	
Competences Expected: AG-C07,AG-C12	
Topics	**Learning Outcomes**
• Relaciones: propiedades, relaciones de equivalencia, órdenes parciales • Funciones: inyectivas, sobreyectivas, biyectivas, composición, inversa • Grafos: definiciones, caminos, ciclos, árboles, conectividad • Teoría avanzada de grafos: planaridad, coloración, redes	• Determinar si una relación es una relación de equivalencia [Aplicar] • Componer funciones y encontrar inversas [Aplicar] • Definir conceptos básicos de grafos [Explicar] • Resolver problemas básicos de grafos [Aplicar] • Analizar propiedades avanzadas de grafos [Evaluar]
Readings : [Ros19], [Sch12]	

Unit 4: Matemáticas Discretas (12 hours)	
Competences Expected: AG-C07,AG-C12	
Topics	**Learning Outcomes**
• Conteo: permutaciones, combinaciones, teorema del binomio • Conteo avanzado: funciones generadoras, principio de inclusión-exclusión • Relaciones de recurrencia: resolver recurrencias simples • Recurrencia avanzada: funciones generadoras, comportamiento asintótico	• Explicar permutaciones y combinaciones [Explicar] • Aplicar técnicas de conteo para resolver problemas [Aplicar] • Definir relaciones de recurrencia [Explicar] • Resolver relaciones de recurrencia simples [Aplicar] • Desarrollar funciones generadoras para problemas de conteo [Crear]
Readings : [Ros19], [Gri03]	

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

********* EVALUATION MISSING ********

10. BASIC BIBLIOGRAPHY

[Gri03] Ralph P. Grimaldi. *Discrete and Combinatorial Mathematics: An Applied Introduction*. 5th. Pearson, 2003.

[Sch12] Edward A. Scheinerman. *Mathematics: A Discrete Introduction*. 3rd. Cengage Learning, 2012.

[LLM18] Eric Lehman, F. Thomson Leighton, and Albert R. Meyer. *Mathematics for Computer Science*. MIT OpenCourseWare, 2018.

[Ros19] Kenneth H. Rosen. *Discrete Mathematics and Its Applications*. 8th. McGraw-Hill Education, 2019.

[Vel19] Daniel J. Velleman. *How to Prove It: A Structured Approach*. 3rd. Cambridge University Press, 2019.