NATIONAL UNIVERSITY OF SAN MARCOS

FACULTY OF MATHEMATICAL SCIENCES

SYLLABUS

1. GENERAL INFORMATION

1.1 Course name	:	Numbers theory
1.2 Code	:	MA281
1.3 Academic Program	:	Scientific Computing
1.4 Credits	:	2
1.5 Prerrequisites	:	MA100. Basic Mathematics. (1^{st} Sem)
1.6 Type of Course	:	Mandatory
1.7 Area of Studies	:	Estudios Generales
1.8 Number of Weeks	:	16
1.9 Hours per week	:	1 HT; 2 HP;
1.10 Cycle	:	I
1.11 Learning modality	:	Face to face
1.12 Academic Period	:	2026-I
1.13 Semester Start Date	:	
1.14 Semester End Date	:	

2. PROFESSORS

Meetings after coordination with the professor

3. INTRODUCTION TO THE COURSE

Number Theory is a fundamental branch of pure mathematics that studies the properties of integers, serving as the foundation for numerous applications in computational mathematics, cryptography, and information theory. This course equips students with the theoretical and algorithmic tools to analyze problems of divisibility, primality, congruences, and Diophantine equations, developing rigorous logical-abstract thinking. The computational focus allows for the implementation and validation of theoretical concepts, laying the groundwork for areas such as public-key cryptography and factorization algorithms.

4. GOALS

- Understand and apply the fundamental principles of divisibility, the Euclidean algorithm, and the fundamental theorem of arithmetic.
- Analyze and solve problems involving linear congruences and systems of congruences using the Chinese Remainder Theorem.
- Study basic arithmetic functions and their applications in the distribution of prime numbers.
- Introduce the concepts of quadratic residues and the law of quadratic reciprocity, understanding their theoretical importance.
- Implement basic numerical algorithms (Euclid's algorithm, primality testing, modular exponentiation) to validate theoretical concepts.

5. COMPETENCES

ABET-1) An ability to identify, formulate, and solve broadly defined technical or scientific problems by applying knowledge of mathematics and science and/or technical topics to areas relevant to the discipline. (Usage)

6. TOPICS

Unit 1: Divisibility and Congruences (16 hours)	
Competences Expected: ABET-1	
Topics	**Learning Outcomes**
• Axioms of the integers: Well-Ordering Principle and Mathematical Induction. • Divisibility: Division Algorithm, Greatest Common Divisor (Extended Euclidean Algorithm) and Bézout's Lemma. • Prime and composite numbers. Fundamental Theorem of Arithmetic. Infinitude of primes. • Congruences: definition, algebraic properties, and residue classes modulo n. • Computational applications: implementation of the Euclidean algorithm and basic primality tests.	• Understand the fundamental axioms of the integers and the proof technique of induction [Familiarizarse]. • Use the extended Euclidean algorithm to compute the greatest common divisor and the Bézout identity [Usar]. • Prove basic properties of prime numbers and congruences, applying them to elementary problems [Evaluar].
Readings : [**Rosen2011**], [**Apostol1976**]	

Unit 2: Diophantine Equations and the Chinese Remainder Theorem (16 hours)	
Competences Expected: ABET-1	
Topics	**Learning Outcomes**
• Linear Diophantine equations: conditions for solvability and general solution method. • Chinese Remainder Theorem for systems of linear congruences with coprime moduli. • Modular arithmetic: the ring $\mathbb{Z}_n$, units, and zero divisors. • Fermat's and Euler's Theorems. Euler's totient function ϕ and its properties. • Applications in basic cryptography: Caesar cipher and foundations of RSA.	• Identify the conditions for the existence of solutions to linear Diophantine equations [Familiarizarse]. • Apply the Chinese Remainder Theorem to solve systems of congruences and calendar problems [Usar]. • Use Fermat's and Euler's theorems to simplify modular exponentiation calculations and analyze basic algebraic structures [Evaluar].
Readings : [**Rosen2011**], [**Burton2011**]	

Unit 3: Arithmetic Functions and Multiplicative Number Theory (16 hours)	
Competences Expected: ABET-1	
Topics	**Learning Outcomes**
• Arithmetic functions: Definition and examples (tau, sigma, phi, mu). • Dirichlet convolution and the Möbius inversion formula. • Perfect, amicable, and abundant numbers. Open conjectures. • Orders and primitive roots modulo a prime. • Discrete logarithms: definition and applications in key exchange protocols (Diffie-Hellman).	• Recognize the properties of the most important arithmetic functions [Familiarizarse]. • Apply Möbius inversion to solve counting problems in number theory [Usar]. • Compute orders and primitive roots, relating them to the cyclic structure of $\mathbb{Z}_p^*$ [Evaluar].
Readings : [**Apostol1976**], [**Nathanson2000**]	

Unit 4: Quadratic Residues and Computational Applications (16 hours)	
Competences Expected: ABET-1	
Topics	**Learning Outcomes**
• Quadratic residues and non-residues. Legendre symbol and its properties. • Law of Quadratic Reciprocity. Proof and applications in computing the Legendre symbol. • Jacobi symbol and its use in probabilistic primality tests. • Introduction to primality testing: Miller-Rabin test. • Integer factorization: Pollard's $p-1$ method and notions of the RSA algorithm.	• Determine whether a number is a quadratic residue modulo a prime using the Legendre symbol [Familiarizarse]. • Apply the Law of Quadratic Reciprocity to compute Legendre and Jacobi symbols efficiently [Usar]. • Implement basic primality testing and factorization algorithms, analyzing their complexity and limitations [Evaluar].
Readings : [**Rosen2011**], [**Hoffstein2008**]	

7. WORKPLAN

7.1 Methodology

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

7.2 Theory Sessions

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

7.3 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.

8. EVALUATION SYSTEM

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

9. BASIC BIBLIOGRAPHY