NATIONAL UNIVERSITY OF SAN MARCOS

FACULTY OF MATHEMATICAL SCIENCES

SYLLABUS

1. GENERAL INFORMATION

1.1 Course name	:	Numerical Analysis I
1.2 Code	:	MA221
1.3 Academic Program	:	Scientific Computing
1.4 Credits	:	3
1.5 Prerrequisites	:	• MA112. Calculus II. (3^{rd} Sem) • MA122. Linear Algebra II. (4^{th} Sem) • AI161. Applied AI. (2^{nd} 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; 4 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

This course introduces fundamental numerical methods for solving mathematical problems through computational algorithms. It develops skills for implementing and analyzing numerical methods in linear algebra, nonlinear equations, and interpolation, with emphasis on error analysis and numerical stability.

4. GOALS

- Understand fundamentals of computational arithmetic and error analysis
- Implement numerical methods for linear systems of equations
- Solve nonlinear equations using iterative methods
- Apply function interpolation and approximation techniques

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: Computational Arithmetic and Error Analysis (20 hours)	
Competences Expected: ABET-1	
Topics	**Learning Outcomes**
• Floating-point arithmetic and the IEEE standard • Sources of error: truncation, rounding, and discretization • Error propagation and condition number • Stability of numerical algorithms • Absolute, relative, and forward/backward error	• Define the different types of errors in numerical computation (rounding, truncation, discretization) [Familiarity] • Explain the principles of floating-point representation and arithmetic according to the IEEE standard [Familiarity] • Calculate the absolute and relative error of an approximation [Usage] • Analyze the propagation of errors in a sequence of arithmetic operations [Usage] • Distinguish between the condition of a problem and the stability of an algorithm [Assessment] • Compute the condition number of a simple problem (e.g., matrix-vector multiplication) [Usage]
Readings : [**Burden2016**], [**Chapra2015**]	

Unit 2: Numerical Linear Algebra (20 hours)	
Competences Expected: ABET-1	
Topics	**Learning Outcomes**
• LU decomposition and its applications • QR decomposition and least squares problems • Singular Value Decomposition (SVD) and its applications • Condition number and numerical stability of linear systems • Iterative methods for linear systems (Jacobi, Gauss-Seidel, Conjugate Gradient)	• Implement LU and QR decomposition algorithms for solving linear systems [Usage] • Explain the geometric and algebraic meaning of the Singular Value Decomposition [Familiarity] • Use SVD for applications such as matrix approximation and principal component analysis [Usage] • Analyze the stability of a linear system using its condition number [Assessment] • Select and implement an appropriate iterative method for solving large, sparse linear systems [Assessment]
Readings : [**Burden2016**], [**Chapra2015**]	

Unit 3: Roots Finding and Basic Numerical Optimization (20 hours)	
Competences Expected: ABET-1	
Topics	**Learning Outcomes**
• Bisection method and fixed-point iteration • Newton's method and secant method for root finding • Convergence analysis and order of convergence • Gradient descent for univariate optimization • Rate of convergence and stopping criteria	• Implement root-finding algorithms (bisection, Newton, secant) and analyze their convergence [Usage] • Classify the order of convergence of iterative methods [Assessment] • Apply gradient descent to solve simple univariate optimization problems [Usage] • Design appropriate stopping criteria for iterative methods [Usage]
Readings : [**Burden2016**], [**Chapra2015**]	

Unit 4: Interpolation and Function Approximation (20 hours)	
Competences Expected: ABET-1	
Topics	**Learning Outcomes**
• Polynomial interpolation (Lagrange, Newton forms) • Spline interpolation (cubic splines) • Least squares approximation • Runge's phenomenon and Chebyshev nodes • Error analysis for interpolation	• Construct interpolating polynomials using different methods [Usage] • Implement cubic spline interpolation [Usage] • Apply least squares method for data fitting [Usage] • Explain Runge's phenomenon and strategies to mitigate it [Familiarity] • Analyze the error of an interpolation scheme [Assessment]
Readings : [**Burden2016**], [**Chapra2015**]	

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