Monsoon 2023

<u>COURSE BROCHURE</u>

Engineering Mathematics in Action

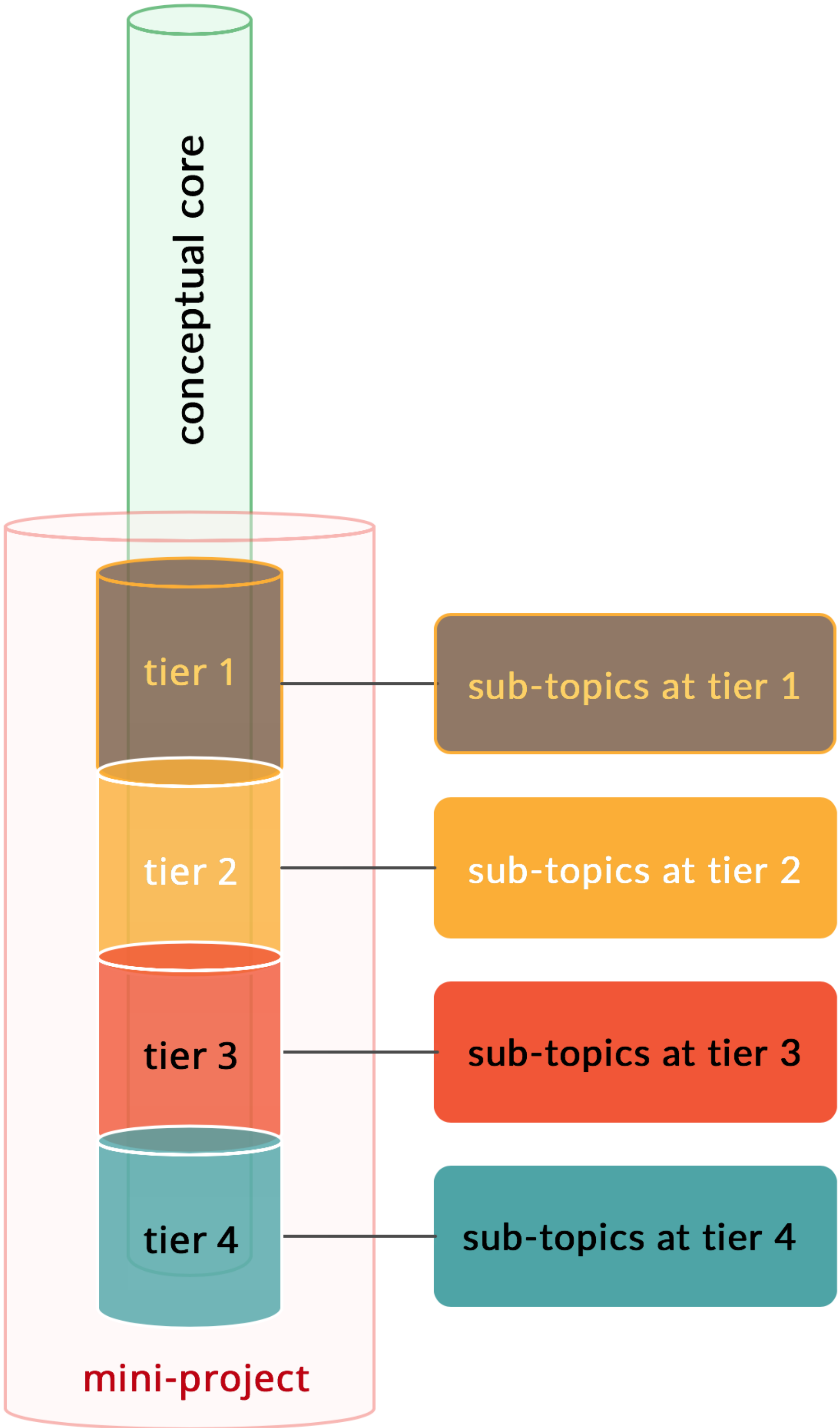
Credits = 3 (Lecture: 2, Laboratory: 1)

Prepared by: *Amrik Sen*

Instructors: *Amrik Sen*, *Vivek Deulkar*

Did you know that your money lives in a matrix?

1 Architecture of Course Learning Unit



2 Syllabus

2.1 Course preamble

This course will cover fundamental aspects of linear algebra and ordinary differential equations from the stand point of basic theoretical knowledge and practical applications. Students will acquire training in foundational concepts. Additionally, they will learn how to use a computer to solve mathematical problems relevant to a broad engineering curriculum. The course is divided into three modules **C1-C3**, each spanning about four-six weeks. Each module comprises a *conceptual core* which is split across sub-modules called *tiers* **T1, T2, T3, ...** . Each tier will cover several related topics that will be discussed over weekly lectures and laboratory classes. There will be two lectures of fifty minutes each and one laboratory class of two hours every week. The course will require completion of two topical mini-projects spread across the semester.

2.2 Pre-requisites

- High school calculus with competency in computing limits of functions, derivatives of functions, and integration of standard functions
- High school geometry and trigonometry
- Some experience with a programming language is desirable (not mandatory)

2.3 Recommended textbooks

- *Linear Algebra with Applications*, by *Otto Bretscher*, fifth edition, Pearson, (2019).
- *Differential Equations and Linear Algebra* by *Jerry Farlow, James E. Hall, Jean Marie McDill, and Beverly H West*, Pearson, second edition (2007).

2.4 Assessments and schedule

The course is divided into two parts comprising two modules on linear algebra and one module on differential equations. The weightage for each of the two parts is approximately 70% and 30% respectively accounting upto a total of 100% points.

- Three written exams: 30% (mid-term) + 20% (two quizzes) = 50% weightage.
- Projects: 25% weightage (project 1) + 25% weightage (project 2).
- The exact dates and venues of the quizzes will be notified in class by the instructor.
- The assessments for projects will be primarily based on student's performance in one-on-one interviews that will test both theoretical knowledge as well as coding skills.
- There is no end semester examination for this course.
- Students must separately pass theory and laboratory section of the course to obtain a pass grade in this course.

2.5 Course Learning Outcomes (CLOs)

Upon successful completion of this course, students should be able to acquire the following skills.

1. Construct matrix representations of systems of linear equations and find their numerical solutions.
2. Compute orthonormal bases of vector spaces both analytically and numerically using a computer algorithm.
3. Compute eigenvalues and eigenvectors of matrices analytically, and interpret the results geometrically.
4. Solve higher order linear differential equations both analytically and numerically.
5. Perform several complex mini-projects involving knowledge and application of a multitude of concepts.

3 Course content and topics

Module C1: Vector spaces, linear independence and bases, systems of linear equations

(i) Introduction to matrices.

(ii) Examples of systems of linear equations and matrices.

(iii) Vector spaces and examples.

(iv) Bases, dimensions, and subspaces.

(v) Linear transformation and subspaces.

(vi) Linear transformation and solutions to systems of linear equations.

(vii) Norms of matrix and vectors, Iteration schemes.

Module C2: Orthonormal basis, eigenvalues and eigenvectors

(i) Vector orthogonalization and matrix factorization.

(ii) Eigenvalues, eigenvectors, and applications.

(iii) Numerical methods to solve eigenvalues and eigenvectors

Module C3: Ordinary Differential Equations (ODEs) and their solutions

(i) Introduction to ODEs.

(ii) Some analytical solutions to ODEs.

(iii) Solutions to higher order ODEs and systems of equations.

(iv) Numerical solutions to ODEs.

Weekly lecture, laboratory and project plan

In this section, a detailed weekly lecture plan is provided. The laboratory classes will require access to a computer/ laptop and PYTHON or MATLAB software. You are encouraged to collaborate with your peers in small groups of 4-5 students in the laboratory. Assessments of projects will be done individually (one-on-one interviews). The course projects will require teams of 4-5 students to collaborate and submit a joint report including both code and analysis. Students will be required to submit project reports through an online submission portal (this will be notified in detail by the instructor). Project reports may be written using Latex or MS-Word but submissions will be considered in pdf format only. For writing equations, students must use a math editor.

Naming convention:
L2.1 refers to 1st lecture of 2nd week.
P2.1 refers to 1st laboratory class of 2nd week.

Week	Lecture Topic	Laboratory Topic
MODULE-CI	Vector spaces, linear independence and bases, systems of linear equations	
Week-1	Introduction to matrices	
L1.1 L1.2 P1.1	Introduction Matrices and vectors: Basics	Introduction to Python for mathematics: Part-I
Week-2	Examples of systems of linear equations and matrices	
L2.1 L2.2 P2.1	Representation of systems of equations in matrix-vector form Representation of systems of equations in matrix-vector form: Engineering applications	Introduction to Python for mathematics: Part-II
Week-3	Vector spaces and examples	
L3.1 L3.2 P3.1	Definitions and examples of vector spaces. Concept of linear independence of vectors.	Python tutorial + introducing arrays + mathematical operations with arrays.
Week-4	Bases, dimensions, and subspaces	
L4.1 L4.2 P4.1	Bases and dimensions of vector spaces. Important subspaces of vector spaces.	Python tutorial + plotting the span of vectors.

Week-5	Linear transformation and subspaces	
L5.1 L5.2 P5.1	Row reduced echelon form, rank of a matrix, linear transformation, image and kernel of linear transformation. Examples on linear transformations and important vector spaces: Part I	Python tutorial + rank + row-reduced echelon form.
Week-6	**Linear transformation and solutions to systems of linear equations**	
L6.1 L6.2 P6.1	Examples on linear transformations and important vector spaces: Part II Solution of system of linear equations by Gauss elimination.	Python tutorial + solving systems of equations. Project-1 launch.
Week-7	**Norms of matrix and vectors, Iteration schemes**	
L7.1 L7.2 P7.1	Matrix norms and vector norms, iterative method(s) for solving linear system using Jacobi method. Iterative method(s) for solving linear systems using Jacobi method, convergence results for general iteration methods.	Python tutorial for iterative methods. Project-1: continuous assessment-part I.

MODULE-C2	Orthonormal basis, eigenvalues, and eigenvectors	
Week-8	**Vector orthogonalization and matrix factorization**	
L8.1 L8.2 P8.1	Gram-Schmidt orthonormalization. Matrix factorization.	Project-1: continuous assessment-part II.
Week-9	**MID-SEMESTER EXAMINATION**	
Week-10	**Eigenvalues, eigenvectors, and applications**	
L10.1 L10.2 P10.1	Definition of eigenvalues and eigenvectors, and their physical and geometrical meaning: Part-I. Definition of eigenvalues and eigenvectors, and their physical and geometrical meaning: Part-II.	Project-1: continuous assessment-part III.

Project 1	**Project 1: due by the end of laboratory class of week-10**	**Project 1: Grand finale (completion of all assessments)**
Week-11	**Numerical methods to solve eigenvalues and eigenvectors**	
L11.1 **L11.2** **P11.1**	**Numerical methods to compute eigenvalues and eigenvectors: Part-I.** **Numerical methods to compute eigenvalues and eigenvectors: Part-II.**	**Python tutorial + coding for power method**
MODULE-C3	**Ordinary Differential Equations (ODEs) and their solutions**	
Week-12	**Introduction to ODEs**	
L12.1 **L12.2** **P12.1**	**Introduction to ODEs with example** **Solutions to linear ODEs with constant coefficients: Part-I**	**Python tutorial on writing ODEs. Project 2 launch.**
Week-13	**Some analytical solutions to ODEs**	
L13.1 **L13.2** **P13.1**	**Solutions to linear ODEs with constant coefficients: Part-II** **Solutions to linear ODEs with constant coefficients: Part-III**	**Python tutorial on numerical solutions to ODEs using in-built Python methods.** **Project-2: continuous assessment-part I.**
Week-14	**Solutions to higher order ODEs and systems of equations**	
L14.1 **L14.2** **P14.1**	**Transformation of higher order ODEs to system of first order ODEs in matrix-vector form** **Solutions to systems of ODEs**	**Python tutorial on numerical solutions to ODEs using in-built methods.** **Project-2: continuous assessment-part II.**
Week-15	**Numerical solutions to ODEs**	
L15.1 **L15.2** **P15.1**	**Numerical solutions of ODEs: Runge-Kutta method: Part-I.** **Numerical solutions of ODEs: Runge-Kutta method: Part-II.**	**Project-2: continuous assessment-part III.**
Project 2	**Project 2: due by the end of laboratory class of week-15**	**Project 2: Grand finale (completion of all assessments)**
Week-16	**No end-semester examination**	