

<b>COURSE TITLE</b>	<b>NUMERICAL METHODS WITH PYTHON</b>
<b>COURSE CODE</b>	<b>01AI0607</b>
<b>COURSE CREDITS</b>	<b>4</b>

**Objective:**

- 1 Numerical methods and AI are primary ways to employ computer powers in latest designing problems. In this course programming Numerical Methods in Python will be focused. This course exposes students to classical topics in Numerical Analysis: The solution of linear and non-linear equations, conditioning, least squares, numerical computation of eigenvalues, interpolation, quadrature, and numerical methods for ODEs. The course will have a focus on the analysis of numerical methods, but also require to use programming software.
- 2 Numerical methods and AI are primary ways to employ computer powers in latest designing problems. In this course programming Numerical Methods in Python will be focused. This course exposes students to classical topics in Numerical Analysis: The solution of linear and non-linear equations, conditioning, least squares, numerical computation of eigenvalues, interpolation, quadrature, and numerical methods for ODEs. The course will have a focus on the analysis of numerical methods, but also require to use programming software.

**Course Outcomes:** After completion of this course, student will be able to:

- 1 Find an approximate solution of algebraic equations using appropriate method.
- 2 Solve system of linear equations and find an eigenvalue using appropriate iterative method.
- 3 Apply various interpolation methods and finite difference concepts.
- 4 Work out numerical integration whenever and wherever routine methods are not applicable
- 5 Find an approximate solution of Ordinary Differential Equations using appropriate iterative method.
- 6 Implement numerical methods in Python and Write efficient, well-documented code in Python.

**Pre-requisite of course:** Knowledge of undergraduate linear algebra and calculus. Python will be used as the primary language.

**Teaching and Examination Scheme**

<b>Theory Hours</b>	<b>Tutorial Hours</b>	<b>Practical Hours</b>	<b>ESE</b>	<b>IA</b>	<b>CSE</b>	<b>Viva</b>	<b>Term Work</b>
3	0	2	50	30	20	0	0

<b>Contents : Unit</b>	<b>Topics</b>	<b>Contact Hours</b>
1	<b>Roots of equation</b> Successive approximation, Bisection method, False-position, Secant, Newton-raphson, Rate of convergence	8

<b>Contents : Unit</b>	<b>Topics</b>	<b>Contact Hours</b>
2	<b>Applied linear algebra</b> Gauss elimination, partial pivoting, Gauss-Jacobi method, Gauss-Seidel method, LU factorization, over method for computation of Eigen values	10
3	<b>Interpolation</b> Finite Differences, Forward, Backward and Central operators, Newton's forward, Backward interpolation formulae, Gauss & Stirling's central difference formulae, Newton's divided difference, Lagrange's interpolation formulae for unequal intervals	12
4	<b>Numerical integration</b> Newton-Cotes integration formulas, trapezoidal rule and Simpson's rules, Gaussian quadrature formulae	8
5	<b>Ordinary differential equations</b> Numerical solution of ordinary differential equations, Euler and Runge-Kutta methods	5
<b>Total Hours</b>		<b>43</b>

#### Suggested List of Experiments:

<b>Contents : Unit</b>	<b>Topics</b>	<b>Contact Hours</b>
1	<b>Practical</b> Develop a Python code to find a root of non-linear equation using Bisection method., Develop a Python code to find a root of non-linear equation using Regula-Falsi method., Develop a Python code to find a root of non-linear equation using Newton-Raphson method., Develop a Python code for Gauss Jordan elimination method., Develop a Python code for Gauss Seidel method., Develop a Python code to compute the interpolation value using Newton's Forward Difference formula., Develop a Python code to compute the interpolation value using Newton's Backward Difference formula., Develop a Python code to compute Lagrange's interpolation formula., Develop a Python code to compute Newton's divided difference formula., Develop a Python code for linear curve fitting., Develop a Python code to implement Trapezoidal Rule., Develop a Python code to implement Simpsons 1/3 and Simpsons 3/8 Rule., Develop a Python code to solve ODE using Runge- Kutta 2nd order and 4th order method	26
<b>Total Hours</b>		<b>26</b>

#### Textbook :

- 1 Applied Numerical Analysis, Gerald, C. F. and Wheatly, Wesley, 6th edition, 2011

#### References:

- 1 Gerald, C. F. and Wheatly, P. O., " Applied Numerical Analysis", 6th Edition, Wesley.

**References:**

- 2 Jain, M. K., Iyengar, S. R. K. and Jain, R. K., "Numerical Methods for Scientific and Engineering Computation", New Age Pvt. Pub, New Delhi
- 3 Chapra, S. C. & Canale, R. P., " Numerical Methods for Engineers " SIXTH EDITION, Mc Graw Hill Publication
- 4 Conte, S. D. and De Boor, C., "Elementary Numerical Analysis", Mc Graw Hill Publisher.
- 5 Krishnamurthy, E. V. & Sen, S. K., "Applied Numerical Analysis", East West Publication
- 6 J. Kiusalaas, Numerical methods in engineering with Python 3. Cambridge University Press, 2013

**Suggested Theory Distribution:**

The suggested theory distribution as per Bloom’s taxonomy is as follows. This distribution serves as guidelines for teachers and students to achieve effective teaching-learning process

Distribution of Theory for course delivery					
<b>Remember / Knowledge</b>	<b>Understand</b>	<b>Apply</b>	<b>Analyze</b>	<b>Evaluate</b>	<b>Higher order Thinking / Creative</b>
15.00	40.00	30.00	15.00	0.00	0.00

**Instructional Method:**

- 1 The course delivery method will depend upon the requirement of content and need of students. The teacher in addition to conventional teaching method by black board, may also use any of tools such as demonstration, role play, Quiz, brainstorming, MOOCs etc.
- 2 The internal evaluation will be done on the basis of continuous evaluation of students in the laboratory and class-room.
- 3 Practical examination will be conducted at the end of semester for evaluation of performance of students in laboratory
- 4 Students will use supplementary resources such as online videos, NPTEL videos, e-courses, Virtual Laboratory

**Supplementary Resources:**

- 1 [https://onlinecourses.nptel.ac.in/noc21\\_ma45/preview](https://onlinecourses.nptel.ac.in/noc21_ma45/preview)
- 2 <https://www.codecademy.com/learn/learn-python>
- 3 <https://www.datacamp.com/courses/intro-to-python-for-data-science>