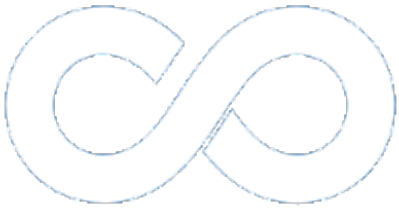
National Research University Higher School of Economics



coursera.org/learn/intro-to-numerical-analysis/home/info

Introduction to numerical analysis

About this Course

Numerical computations historically play a crucial role in natural sciences and engineering. These days however, it’s not only traditional «hard sciences»: whether you do digital humanities or biotechnology, whether you design novel materials or build artificial intelligence systems, virtually any quantitative work involves some amount of numerical computing . These days, you hardly ever implement the whole computation yourselves from scratch. We rely on libraries which package tried-and-tested, battle-hardened numerical primitives. It is vanishingly rare however that a library contains a single pre-packaged routine which does all what you need. Numerical computing involves assembling these building blocks into computational pipelines. This kind of work requires a general understanding of basic numerical methods, their strengths and weaknesses, their limitations and their failure modes. And this is exactly what this course is about. It is meant to be an introductory, foundational course in numerical analysis, with the focus on basic ideas. We will review and develop basic characteristics of numerical algorithms (convergence, approximation, stability, computational complexity and so on), and will illustrate them with several classic problems in numerical mathematics. You will also work on implementing abstract mathematical constructions into working prototypes of numerical code. Upon completion of this course, you will have an overview of the main ideas of numerical computing, and will have a solid foundation for reading up on and working with more advanced numerical needs of your specific subject area. As prerequisites for this course, we assume a basic command of college-level mathematics (linear algebra and calculus, mostly), and a basic level of programming proficiency. Do you have technical problems? Write to us: coursera@hse.ru

More

Level	Intermediate
Commitment	7 weeks of study, 5-6 hours/week
Language	English Volunteer to translate subtitles for this course
Hardware Req	For the assignments, you will need a working installation of the Python programming language.
How To Pass	Pass all graded assignments to complete the course.

10 videos, 1 reading

1. **Video:** Introduction.
2. **Ungraded Plugin:** Pre-survey on HSE online courses
3. **Reading:** About the course
4. **Video:** A simple worked example.
5. **Video:** Machine arithmetics. Representation of real numbers.
6. **Video:** Machine epsilon. Over- and underflow.
7. **Video:** A crude estimate of the machine epsilon.
8. **Video:** Systems of linear equations. Cramer's rule.
9. **Video:** Gaussian elimination.
10. **Video:** LU decomposition: the matrix form of the Gaussian elimination.
11. **Video:** When does the Gaussian elimination work?
12. **Video:** LU decomposition with pivoting. Permutation matrices.
13. **Peer Review:** LU pivoting of a square matrix.

Graded: Solve a quadratic equation.

12 videos

1. **Video:** Introduction.
2. **Video:** Sensitivity of a linear system.
3. **Video:** Vector norms.
4. **Video:** Matrix norms.
5. **Video:** Common matrix norms.
6. **Video:** Sensitivity of a linear system. Condition number.
7. **Video:** Cholesky decomposition.
8. **Video:** Banded matrices. Thomas algorithm.
9. **Video:** Shermann-Morrison formula.
10. **Video:** QR decomposition.
11. **Video:** Constructing the QR decomposition: Householder reflections.
12. **Video:** Constructing the QR decomposition: Givens rotations

Graded: QR decomposition of a matrix

11 videos

1. **Video:** Solving non-linear equations.
2. **Video:** Localization of roots. Bisection.
3. **Video:** Fixed-point iteration.
4. **Video:** Aside: convergence rates and related technicalities.
5. **Video:** Back to the fixed-point iteration.
6. **Video:** Fine-tuning the fixed-point iteration.
7. **Video:** Newton's iteration.
8. **Video:** Multiple roots. Modified Newton's method.
9. **Video:** Inverse quadratic interpolation.
10. **Video:** Roots of polynomials.
11. **Video:** Roots of polynomials: the companion matrix.
12. **Peer Review:** Solving non-linear algebraic equations.
13. **Ungraded Plugin:** Survey

7 videos

1. **Video:** Large-scale systems of linear equations.
2. **Video:** Simple iteration for a linear system. Jacobi iteration.
3. **Video:** Convergence criteria for simple iteration.
4. **Video:** Seidel's iteration.
5. **Video:** Successive over-relaxation.
6. **Video:** Canonic form of two-step iterative methods for linear systems.
7. **Video:** Variational approaches: minimum residual method.
8. **Peer Review:** Iterative methods for linear systems.

9 videos

1. **Video:** Interpolation and approximation. Modelling of data.
2. **Video:** Linear least squares problem.
3. **Video:** Ordinary least squares: the normal equations.
4. **Video:** Ordinary least squares: QR decomposition of the design matrix.
5. **Video:** Global polynomial interpolation.
6. **Video:** Lagrange interpolating polynomial.
7. **Video:** Quantifying interpolation errors. Runge phenomenon.
8. **Video:** Chebyshev nodes.
9. **Video:** Interpolation of the Runge function.

Graded: Interpolation and approximation. Modeling of data.

13 videos

1. **Video:** Numerical derivatives.
2. **Video:** Numerical derivatives: finite differences.
3. **Video:** Truncation and roundoff errors: an interplay.
4. **Video:** Higher order schemes.
5. **Video:** Richardson extrapolation.
6. **Video:** Integration: numeric quadratures.
7. **Video:** Convergence rates of simple quadratures.
8. **Video:** Simple geometric quadratures: Trapezoids, Simpson's rule and all that.
9. **Video:** Error bounds for quadratures. Romberg extrapolation.
10. **Video:** Integrals with singularities.
11. **Video:** A check of convergence.
12. **Video:** Recap: Newton-Cotes vs Gaussian quadratures.
13. **Video:** Gaussian quadratures.
14. **Peer Review:** Numerical calculus: derivatives and integrals.

7 videos

1. **Video:** Initial value problem for an ODE. Discretization.
2. **Video:** Approximation and convergence.
3. **Video:** Truncation error or Euler-like schemes.
4. **Video:** Runge-Kutta methods.
5. **Video:** Asymptotic stability of ODEs. Stiffness.

6. **Video:** Linear Multistep methods.
7. **Video:** Zero-stability of linear multistep methods.

Graded: Initial value problem for ordinary differential equations.