Lecturer	**:** Christina Christara (`ccc@cs.toronto.edu`)
Lectures	**:** Tuesday 6-9 PM, Room MP 134
Tutorial	**:** Some lecture times will be used for tutorials
Office Hours	**:** Tuesday 4:30-5:30pm, Room BA 4226, other hours by appointment
Textbook	**:** Michael Heath, Scientific Computing: an introductory survey, SIAM 2018 available from the SIAM website, cheaper if you become student member, see details in course website below equivalent to same book McGraw-Hill Inc. 2002, custom-copy used in past years
alternative	: Uri Ascher and Chen Greif, A first course in Numerical Methods, SIAM 2011 (e-book on library)
Web site	**:** `http://www.cs.toronto.edu/~ccc/Courses/436.html`
Bulletin board	**:** `https://` (given after first week)

Aims of course

Formulate numerical methods for approximation, integration, eigenproblems and ODEs.
Evaluate numerical methods with respect to their convergence, stability, and efficiency.
Develop and practice computer skills in implementing numerical methods efficiently on the computer.
Introduce concepts/techniques, such as adaptivity and event detection, to develop robust, reliable and flexible numerical methods.
Use high level software for studying numerical methods.

Skills / Knowledge testing in the course

Apply basic principles, not recall lecture notes in detail
Problem recognition
Method recognition
Apply a given method correctly
Solve a numerical problem efficiently and reliably using mathematical software.
Judge the quality and efficiency of the numerical results.

Prerequisite Mathematics and Numerical Analysis

Ability to handle notation and to do algebraic manipulation

Induction

Calculus including differentiation and integration of polynomial, trigonometric, exponential, logarithmic and rational functions, continuity, limits, graphs of functions, Taylor series, Rolle's theorem, mean-value theorem, de l' Hospital's rule, some exposure to multivariate differentiation, etc.

Elementary Linear Algebra including
Matrix and vector addition and multiplication, elementary row operations, linear (in)dependence, inverse matrix, etc.

Numerical Linear Algebra (such as **CSC336** or **CSC350**) including
Linear solvers for banded matrices and Nonlinear equations solvers

Computational methods: Understanding of round-off error, computer arithmetic, etc.

Programming: knowledge of some programming language, such as MATLAB, python, FORTRAN or C.

Computer accounts

You will get (or have already) a computer account on the Teaching Labs (CDF) Unix system. Consoles/workstations are located in the Bahen building. You must log-in frequently and read mail, news and other messages relating to the course through your account.

Marks distribution

CSC436

Assignment 1	Tue 3 Feb	16%
Test 1	Tue 24 Feb	25%
Assignment 2	Tue 10 Mar	17%
Test 2	Tue 24 Mar	25%
Assignment 3	Mon 6 Apr	17%

• Must get at least 30% in each of the tests, in **each** of the other assessments; can't skip any
• Must get at least 33% average in the computing parts of the assignments.
• **Term tests**: Calculators are the only aids permitted.

Problem sets / Computer assignments

problem sets: please write as clearly as possible.

Indicate your last (family) name by capitalisation or underlining in the front page of your paper.

computer assignments: don't leave it to the last minute - think of the following

- the machine being down, when you need it. final listing.
- accidentally deleting an important file.
  overcome this by using backup procedures (for the source and data files only).

The above are not good reasons for extension of the assignment due date.

Late assignment policy

Assignments are due the day posted, the time posted. Assignments submitted late have a reduction of marks based on the maximum total marks the assignment could get, had it been submitted on time (and not on the total marks the assignment actually got). Each day costs 10%, to a maximum of 2 (two) days. Assignments submitted later than 2 days after the due date do not receive any marks. If applicable, weekends and holidays count as regular days for the purpose of late assignment policy.

Topics to be covered

- Interpolation (Ch. 7)
  - Polynomial interpolation - Weierstrass theorem
  - Monomial basis, Lagrange basis, Newton's divided differences
  - Existence and uniqueness of interpolating polynomial
  - Error of polynomial interpolation
  - Evaluation of a polynomial - Horner's rule
  - Polynomial interpolation with derivative data - Hermite interp.
  - Monomial basis, Lagrange basis, Newton's divided differences
  - Existence and uniqueness of Hermite interpolating polynomial
  - Error of Hermite polynomial interpolation
  - Problems with polynomial interpolation - Runge's function
  - Piecewise polynomial interpolation - splines
  - Basis for piecewise polynomials
- Integration -- Quadrature (Ch. 8)
  - Simple quadrature rules - rectangle, midpoint, trapezoidal, Simpson's, corrected trapezoidal
  - Gauss quadrature rules
  - Compound quadrature rules
  - Romberg integration
  - Adaptive integration
  - Infinite integrals, singularities
- Ordinary Differential Equations (Ch. 9)
  - Initial Value Problems and Boundary Value Problems
  - Stability of ODEs and of methods for ODEs
  - Euler's method
  - Implicit methods, backward Euler's and trapezoidal method
  - Runge-Kutta methods
  - Taylor's series methods
  - Linear Multistep Methods
- Least Squares Approximation (Ch. 3)
  - Least squares problems - data fitting
  - Inner products and norms of functions
  - Normal equations, QR factorisation, Gram-Schmidt algorithm
  - Orthogonal and orthonormal polynomials
- Computing eigenvalues and eigenvectors (Ch. 4)
  - The power method; the QR iteration

References

Michael Heath
Scientific Computing: an introductory survey
SIAM 2018 (or McGraw-Hill Inc. 2002)

Uri Ascher and Chen Greif
A first course in Numerical Methods
SIAM 2011 (e-book on library)

Richard L. Burden and J. Douglas Faires
Numerical Analysis
Brooks/Cole

David Kincaid and Ward Cheney
Numerical Analysis
Brooks/Cole

James Epperson
An introduction to Numerical Methods and Analysis
Wiley 2003

Samuel D. Conte and Carl de Boor
Elementary Numerical Analysis
SIAM 2018 (also McGraw-Hill Inc.)

L. W. Johnson and R. D. Riess
Numerical Analysis
Addison Wesley

G. Dahlquist and A. Bjorck (trans. N. Anderson)
Numerical Methods
Prentice Hall

J. Stoer and R. Bulirsch
Introduction to Numerical Analysis
Springer Verlag

Academic integrity

Assignments, homeworks and exams must be your own individual work and using only course materials. While students at your level are well aware of what academic integrity means, please note that violating academic integrity includes more things than presenting others' work as one's own. For example, *not taking reasonable measures to protect your work from being plagiarized by others is also a violation of academic integrity*. This is becoming particularly important now that so many things are online.

You should *never post anywhere or share with anyone* assignments, exams, questions or solutions, *even after the deadline*.

Additional information

Assignments will be submitted electronically; details to be given with each assessment.

Assignments will be preferably typed in latex. A template is given in the course website. Other document processors are acceptable, as long as they produce pdf output. If an assignment is *very cleanly* handwritten and scanned *on a proper scanner* as a single pdf file, and *not photographed*, then it is also acceptable. Photographed assignments will receive 0 marks.

Tests/Exams will be handwritten and in-person.

Must get at least 30% in **each** of the assessments; can't skip any

Lecture and tutorial times may be used interchangeably.

I will hold in-person office hours in my office room (masks welcome) at default times posted, or at other mutually agreeable times; only one student at a time.
Office hours will also be available remotely with pre-arrangement, either at default times posted, or at other mutually agreeable times. Office hours will be for individual students, not for a group of students.

Presentation of assignments

General
Include your name and student id in the front page and underline last name. Use font size 12 or larger. Use fixed width fonts (e.g. Courier) for code and output. *Never* use dark background, for anything.

Tables and code output
Always align output with an appropriate format statement. (Align to match equivalent order digits.) Use exponential format for very large (e.g. condition numbers) numbers and very small numbers (e.g. errors, residuals) Use integer format for number of iterations, grid sizes, etc. Always use headers for columns in tables.

Plots
When we say plot quantity A versus B, we mean A is in the vertical (y) axis and B in the horizontal (x). Always use captions for plots/figures, and proper diacritical marks and legends when drawing more than one line.

Submission
Do NOT submit zip, rar and similar files on MarkUs. Only submit pdf, image (eps, png, etc), text (incl. code, latex), etc.

Other
Do not use any symbolic computation, such as symbolic differentiation, etc.