CONCORDIA UNIVERSITY
Department of Computer Science and Software Engineering
COMP361/5611 - Numerical Methods
Course Outline - Winter 2018

Objective:
Numerical methods are important in Science and Engineering. Computer Scientists should have a basic knowledge of such algorithms. The purpose of this course is to introduce some elements of Numerical Analysis that are fundamental to almost all Scientific computations.

Schedule:
Instructor: Eusebius Doedel
Office Hours: Monday 16.00 – 17.00 EV3.285
Lecture: Monday 17:45 – 20:15 H1070
Tutorial: Monday 20:30 – 21:20 H1070

Outline:
The following topics are covered: Vector and matrix norms, Gauss elimination, LU-decomposition, row-pivoting, condition number, residual correction, nonlinear equations, iterative methods, Newton’s method, convergence analysis, polynomial interpolation, numerical differentiation, best approximation, orthogonal polynomials, numerical integration, Gauss quadrature, and (time permitting) spline interpolation, discrete least squares, numerical solution of ODEs, stability of difference methods, stiff ODEs, boundary value problems for ODEs, and numerical solution of PDEs (diffusion problems).

Prerequisites:
COMP232 and COMP249. Knowledge of linear algebra (vectors and matrices) and calculus (derivatives and integrals) is essential.

Course Material:
The Lecture Notes are available on the course web page given below. The Lecture Notes contain in a concise form the basic material covered in the course. Students are expected to thoroughly understand the Lecture Notes, and to be able to apply the material to simple problems. Students are responsible for all material covered during the lectures, even if the material does not appear in the posted Lecture Notes. There is a strong correlation between class attendance and performance in this course. A useful elementary reference is ”Numerical Computing with MATLAB”, by Cleve Moler, which can be downloaded from http://www.mathworks.com/moler/. Note, however, that there is no formal text book in this course. Students wishing to learn additional material beyond the requirements of the course, are encouraged to borrow appropriate books from the library or from the instructor. However, such supplementary reading is not necessary to prepare for exams.
Course Web Page:
The course web page
http://users.ensc.concordia.ca/~doedel/courses/comp-361/
is used for posting the Lecture Notes, assignments, and eventual announcements. Students are strongly
encouraged to solve the exercises in the Lecture Notes, which are representative of actual examination
problems. These exercises can be discussed in the tutorial, during office hours, and in class. Solutions
to selected exercises will also be posted.

Tutorials:
Tutorials are used to discuss problems related to the lectures and assignments.

Assessment:
There will be up to five assignments, two term tests, and a final examination. Term Test 2, which
will be near the end of the term, will examine a student’s knowledge of all lecture and assignment
material covered so far during the term. Based upon the student’s performance on both term tests and
the assignments, a provisional course letter grade will be assigned, based upon the following weighting
scheme: Assignments: 10%, Term Test 1: 20%, Term Test 2: 70%. However, if the result of the Term
Test 2 is better than the result of Term Test 1 (as a percentage out of 100) then the weight of Term
Test 1 will be shifted to Term Test 2.

Students are then given the option of accepting the provisional course letter grade, in which case they
will not write the final exam, or not accepting the provisional course letter grade, in which case they
must write the final exam. There will be a strict deadline for acceptance of the provisional course letter
grade. If no acceptance is received by the deadline then the student must write the final exam.

If a student chooses to write the final exam then the course letter grade will be based on the following
weighting scheme: Assignments: 10%, Term Test 1: 10%, Term Test 2: 20%, Final Exam: 60%. However, if the result of the Final Exam is better than the result of Term Test 1 (as a percentage out
of 100) then the weight of Term Test 1 will be shifted to the Final Exam. Moreover, if the result of
the Final Exam is better than the result of Term Test 2 (as a percentage out of 100) then the weight
of Term Test 2 will be shifted to the Final Exam.

Note that there is no standard relationship between numerical percentages and letter grades.

Assignments:
Assignments combine theoretical problems and programming exercises. Programming may be done in
any appropriate language such as Matlab, Python, Fortran, C, Java, or C++, although use of Matlab
or Python is encouraged. Assignments can be submitted in class when the assignment is due, or be put
in the instructor’s assignment box by the given deadline. The assignment box is located near (but not
inside) the Computer Science reception area.

Discussing the assignments with other students is encouraged, but it is very important that you do the
actual work yourself. The assignments constitute an essential learning experience without which it may
be difficult to pass the examinations.
CEAB Graduate Attributes:

As part of either the Computer Science or Software Engineering program curriculum, the content of this course includes material and exercises related to the teaching and evaluation of graduate attributes. Graduate attributes are skills that have been identified by the Canadian Engineering Accreditation Board (CEAB) and the Canadian Information Processing Society (CIPS) as being central to the formation of Engineers, Computer Scientists and Information Technology professionals. As such, the accreditation criteria for the Software Engineering and Computer Science programmes dictate that graduate attributes are taught and evaluated as part of the courses. The following is the list of graduate attributes covered in this course, along with a description of how these attributes are incorporated in the course.

Graduate attributes for COMP361 are:

Attribute 1: Knowledge-base: Knowledge of a wide array of fundamental numerical methods used in Science and Engineering, as stated in the course description.

Indicator 1.1: Knowledge base of mathematics

Attribute 2: Problem analysis: Use a wide array of basic numerical methods to model and analyze complex problems in order to establish the requirements and constraints on their design, implementation and deployment solutions.

Indicator 2.1: Problem identification and formulation

Course Learning Objectives:

Introduce the students to basic numerical techniques that are fundamental to Scientific and Engineering computing.

Make the students aware of the complexity and the limitations of numerical algorithms.

Teach the students basic concepts and analytical techniques that allow the determination of key properties of numerical algorithms.

Prepare the students for more advanced courses in Scientific and Engineering computing.