MATH 566 - Introduction to Numerical Analysis

Syllabus

UNC CH - Fall 2017

Course info
TuTh 9:30AM-10:45AM
Phillips Hall 383

Instructor Info
Dr. Simone Rossi
simone.rossi@unc.edu
Phillips Hall 396

Office hours in Phillips Hall 396:
WTh 2-3pm and by appointment

Teaching Assistant Info
Robert Hunt
huntrl@live.unc.edu
Office hours: TBA

PREREQUISITE: MATH 383/283/233/232 or permission of instructor

COURSE DESCRIPTION

At its beginning mathematics was developed as a method of solving practical problems giving sufficientely accurate results: early practical problems required to compute the volume of granaries, the area of fields, the amount of food required for livestock. As time passed, mathematics expanded under the need of solving practical problems in fields such as mechanics and astronomy. During its development, new methods were introduced to increase the accuracy of the calculations as well as to reduce the calculation times. Only in the last century applied mathematics started to be considered as a separate discipline.

In this course we will study some of the algorithms needed to solve these practical problems. While some algorithms are relatively recent (developed in the last century), others date back to ancient mathematics. In particular we will study how to compute roots of nonlinear functions, definite integrals, interpolating polynomials, matrix inverses and more. Our approach will focus on learn-by-doing, practising examples and critical thinking.

The subject of numerical analysis has always being entagled with computing, in the sense of doing mathematical calculations. Nowadays, we can use digital calculators and computers to solve difficult problems in a short amount of time. The interface we use is called programming. Therefore, to be a good numerical analyst you need to know how to program. For this reason we will take an different approach for this class, where you will spend most of the class time in front of a computer actually doing numerical analysis. You will be able to work out most of the theory in the assignments. Although the class assume you have some prior experience in programming, we will try to cover the basics of MATLAB programming: proficiency will come with practice.

REQUIRED EQUIPMENT

As we will have extensive in-class coding activities, you are required to bring your laptop to class fully charged. Your laptop is required to have MATLAB 2017a or a newer version. You can get MATLAB following the link: http://software.sites. unc.edu/software/matlab/. We will test your MATLAB installation during the second class. It is your responsability to make sure that your laptop and MATLAB are fully functioning. No excuses will be made for issues with your laptop/MATLAB installation after the second class. No excuses will be made for issues with your laptop/MATLAB installation after the second class. You are held responsible for any problems with your laptop. Recall that ITS Service Desk can make sure that the software is installed correctly.

You can check the minimum laptop specifications required by UNC at http://cci.unc.edu/new-students/minimum-laptop-requirement/.

UNC Policy: *While undergraduate students are not required to purchase a laptop through the Carolina Computing Initiative, they are required to have a laptop that meets the or exceeds the minimum laptop specifications for their class. Please note that while best efforts will be made to support students with non-CCI laptops, the highest level of support can only be extended to students who purchase a CCI laptop package. Specifically, hardware support and access to the pool of loaner laptops are provided only for students who own a CCI model laptop.*

TOPICS

Given the programming-focus approach of this course and the hetorogeneity of the programming abilities of its participants, the content of the course is subject to change based on students learning and understanding.The following is a list of possible topics that will be covered in this course:

1. Error Approximations: Machine error, Order of convergence
2. Root finding: bisection method, Newton's method, fixed point iterations, systems of nonlinear equations
3. Interpolation and Integration: Taylor/Lagrange/Chebyshev polynomials, piecewise interpolation, least squares interpolation, simple quadrature rules, interpolatory quadratures, Monte Carlo method.
4. Ordinary Differential Equations: forward and backward Euler, trapezoidal and midpoint rule, multistep methods, Runge-Kutta methods, Rush-Larsen methods, Systems of ODEs
5. Linear Systems: LU decomposition, Gaussian Elimination, Cholesky factorization, simple iterative methods
6. * Optimization: Unconstrained optimization, Lagrange and penalty methods
7. * Partial Differential Equations: Finite Difference Method, Finite Element Method

Topics signed with a * may not be covered.

COURSE GOALS

The goal of this course is to introduce students to some of the techniques, practices, and problems that characterize ancient and modern scientific computing. The course is about how to solve real-world problems. We will often focus on examples from science and engineering. The students will be challenged to solve mathematical problems with the aid of programming. The students will discuss their solution, analyzing data and the methods used. The students will write up reports to record their findings.

After successfully completing this course, you should be able to:

1. identify sources of error in scientific programming
2. write functional and well-documented code in MATLAB
3. Articulate what a mathematical model is
4. Analyze algorithms
5. learn how to produce informative scientific data (figures/tables)
6. learn how to incorporate such data into well-written and persuasive scientific reports

The course will contain hands-on activities and projects. This should help you develop secondary skills such as

1. abstract thinking
2. working from first principles
3. working in a team
4. communicating your ideas

ACTIVITIES

Since learning is an active process, we'll use a hands-on approach. In class the students will work on exercises. The instructor, the teaching assistant and the students themselves will be the guides through the exercises.

Note: Some of the instructions for the exercises will not be detailed on purpose. In real life, it's unlikely that you will be given you step-by-step instructions. You need to develop the skill to tackle this kind of situations.

Typical activities will be

- In-class coding activities

- Group work and discussions
- Quizzes
- Projects

In-class and out-of-class assignments will emphasize oral and written communication.

DIGITAL ETIQUETTE

This course will require you to use your laptop during class time. Please be respectful of your classmates and restrict your use of digital devices to course content. Please be respectful of your own learning and realize that those around you will be distracted if you are off-task. If I see that you or your peers are distracted, I will ask you to put your devices away or to leave (temporarily) the class. There will be times when you have completed your work or answered a poll question, but your peers have not. I ask that you assist your peers when appropriate or use the time to review your notes while you wait. You will learn more if you concentrate on the course while you are in class and your classmates will thank you for not impeding their ability to learn. If you must answer a text or a phone call, please step out of the room and return once you have completed your conversation.

Matlab: Homework problems and in-class activities require basic programming skills. This is not a programming class, so it will be assumed that you have some basic programming knowledge. You are asked to use Matlab for the computer assignments. Access to Matlab is available for UNC students free of charge. There are various online resources which teach basic Matlab programming. In case you do not know how to work with Matlab, you should go over some of these resources quickly. Links to several online resources will become available on the course webpage.

REFERENCE MATERIAL

We will follow the material contained in several books and in notes from other courses. Although any book will do, the following two references are available online through the UNC libraries and are explicitely tailored to MATLAB programming

- Scientific computing with MATLAB and Octave. Quarteroni A., Saleri F., Gervasio P. 2014
- Programming for computations -- MATLAB/Octave : a gentle introduction to numerical simulations with MATLAB/Octave. Linge S. Langtangen H.P. 2016 (There is no chapter on linear systems)

Please print the chapters/section required for your assignments. Reading from paper gives you more focus.

The typical (old) standard reference for an introductory course in numerical analysis is

- An introduction to numerical analysis, Atkinson, K.E. 1989

There are other (modern) books on numerical analysis which I can point you to under request.

EXPECTATIONS

Learning is not a spectator sport. Fundamentally, the responsibility to learn is yours and yours alone. For learning to happen in any course, you must take an active role in the process. For our class, you are expected to come to class 'prepared' and 'ready to learn,' which requires you 'to read' and 'to study' the assigned reading 'before' you come to class. Being prepared for class enables you to construct a knowledge base on which subsequent learning rests. During class you will be engaging in exercises and coding activities.

The class will revolve around the idea of productive failure: mistakes are a sign that you are challenging yourself. You will be asked to work outside your comfort zone. The class should be a supportive environment where you can take "risks", getting things wrong without the fear of being judged. Here is a list of concrete expectations:

- attend class
- do all homework/assignments
- participate actively in class activities

- ask questions whenever something is not clear
- volunteer to answer question other students may have
- be courteous and supportive of everyone in the class
- help create a classroom that is supportive and a respectful place to learn.

Important: by the end of the course, you are expected to know all the content of the course, top to bottom. The material contained in the reading assignments must be learned, even if we did not cover the material in class! Most of the theoretical part requires only first-year calculus, so you should not have any difficulties. If you do encounter difficulties get in touch with your instructor or with the TA.

ATTENDANCE

UNC Policy: *Regular class attendance is your obligation, and you are responsible for all the work, including tests and written work, of all class meetings. No right or privilege exists that permits you to be absent from any class meetings except for excused absences for authorized University activities or religious observances required by the your faith. If you miss three consecutive class meetings, or miss more classes than the course instructor deems advisable, the course instructor may report the facts to your academic dean.*

Built into the philosophy of this class is the idea that we help one another to learn. As a consequence, your daily attendance in class is very important, not just for your own benefit, but for that of your peers. I will take role every day and verify attendance. You are allowed to miss up to three days with no penalty. If you miss more than three but no more than five days a penalty up to -5 points may be added to your final grade. After six missed classes I reserve the right to drop you from the class.

Those of you who are motivated to take this class, but cannot participate regularly, should meet with me to work out a solution together.

Absences: If you miss one class/assignment for legitimate reasons (e.g. death in the family, medical emergency, etc.) talk to me and we'll work out something equitable. If you need to be absent for some planned family or medical reason, you should contact me in advance. If an emergency arises, contact me as soon as possible after the emergency has passed. Student athletes who need to miss class for games should let me know of this as early as possible.

ASSIGNMENTS

Remember, for this course you are required to work 9 hours per week out-of-class.

There will be 3 types of assignments:

1. **Reading assignments** ($\sim$ 2 hours/week): You will have material to read before each class. Please print the chapters/sections required for you assignments, as reading from paper gives you more focus. To ensure you do the assignment, we will take a short quiz at the beginning of the class (3 or 4 questions max). The quiz will be randomly graded. The randomness will be decided after the quiz, opening a book: if one of the pages is an integer multiple of 5 the quiz will be graded. If more than one answer is correct, you will receive +1 point bonus added to your final grade. If none of the answers is correct you will receive a penalty of -1 in your final grade.

2. **Quiz** ($\sim$ 1 hour/week): I will provide you with quizzes for each topic. These are meant to help you track your progress and evaluate yourself. You will be able to take them multiple times and they will not be graded. Nonetheless, you are required to take them. For every quiz missed you will get a penally of -1 point on your final grade.

3. **Projects** ($\sim$ 6 hours/week): During the course you will work on 4 projects. The idea is to replicate the process of publishing scientific papers. You will have about 3 weeks to prepare each project. I will give you a (simple) problem that you need to solve numerically. You will prepare a scientific report on that problem, describing the numerical methods used and your result. You will also critically discuss your findings. After the initial submission of your report, we will review your work and give it back with a temporary grade and a feedback. After receiving it back, you will have one week to improve it and submit a second version. A cover letter highlighting the updates in your report will also be submitted with the second version.

   (a) There will be no instructions on how to solve the problem numerically. No code will be evaluated. I will provide you with a rubric showing how we will evaluate of your work.

(b) The grading system is based on the following table

Result	Points Earned
accepted	15
accepted with minor revision	10
accepted with major revision	5
not accepted/not submitted	0

(c) You may carry out the project alone or in groups. For now, the maximum group size is of 2 people. We may change this number during the course. People in the same group will get the same number of points.

(d) You cannot be part of the same group more than twice.

(e) Late submissions will be accepted, but a penalty of -2 points/day will be applied.

(f) We will correct your reports as fast as we can. Understand that we will have a lot of work to do to correct your reports.

(g) The same deadline for the second revision will apply to everyone. We will set the deadlines for the second revisions when we will give back to you the first decision (grade).

(h) You can expect us to spend about 15 minutes for evaluating your initial report and about 5 minutes for the revision. Write your reports accordingly. Guidelines will be provided. (I want you to know the average amount of time we will use to correct your work, as we are trying to also mimic the grant review process. Grant reviewers have to evaluate a large number of grants in a short period of time. Under these circumstances, it's clear that one needs to adapt his own writings to the reviewers.)

(i) Projects will be submitted in pdf format. Microsoft Word documents will not be accepted. You are encouraged to use either plain LaTeX or LyX.

Note: generally the assignments should not take longer than the required 9 hours. You may actually need much less than that. Remember that working for a long time does not translate to good grades. You need quality over quantity. Be focused while you study and the assignments will take less time to complete and will be easier.

The 9 hours do not include the time you may need to catch up with something, from either the coding or theoretical perspective.

EXAMS

You will not be allowed to use any electronic device. You will not be allowed to use any notes.

As you will have a hard to study all the material of all the courses you have taken this semester during the last week before the exams, we will do smaller tests during the semester. The final exam will help you getting a higher grade in case you are not happy with the scores of the tests.

Single Topic Tests:

There will be 5 single topic tests spread throughout the semester. Some tentative dates for these tests are already in the calendar, but the final dates will be announced at least 10 days in advance. The tests will have the format of multiple choice questions or true-or-false questions. You will have about 1 hour to complete each of them. The points taken in the test will be converted in the following way:

1) points to be added to the final grade = points taken in the test / number of questions in the test * 10.

Rounding will be applied after taking the sum of the points on all tests.

Final Exam:

There will be also a final exam. The final exam will be composed of 5 tests. Each test will be on a single topic. The score of each test will be evaluated always as

2) points to be added to the final grade = points taken in the test / number of questions in the test * 10.

If the points taken in 2) are greater than the points in 1), I will add to your final grade such difference. The idea is that if you failed in one or more of the previous tests you have the chance to do better. Note, however, that you will have 3 hours in the final exam. It means about 36 minutes for each test. As your time will be very limited, you should choose to retake only the tests for which you believe you can do better. I discourage you from attempting to solve all the tests: again focus on quality rather than quantity. I designed the final exam to help you recover from possible bad performances during the semester.

GRADES

Your final grade is composed by the following components

Category	Value	Ways to Earn Points
Reports/Projects	60	Prepare a report for each project (max 15 points each)
Single Topic Tests	50	Answer to multiple-choice questions (max 10 point each)
In-class quizzes	random	Take quiz in class, grading will be random (min -1 point each, max 1 point each)
Final Exam	50	Mutually exclusive with single topic tests.

Maximum Points you can get = 110 + random

Final grades are computed as follows:

F	D	D+	C-	C	C+	B-	B	B+	A-	A
<54	55-59	60-64	65-69	70-74	75-79	80-84	85-89	90-94	95-99	≥ 100

GETTING HELP

If you need any help with the class you have my support as well as the support of the teaching assistant (TA). You are welcome to stop by during regular office hours. If you cannot make it, try to schedule an appointment with me or with the TA. Also remember to leverage your peers, the other students attending the class, as they can also offer you great support.

COURSE FEEDBACK

As you will be given feedback during the semester, you should give some feedback to the instructor and to the teaching assistant. Although I'll regularly ask you about it, you should express any concern you may have about the class as soon as you have them. For example, you may feel the class is not challenging, or it is too hard, or the material provided is not good, or the assigments take too much time to complete. Please, let me know so that I can improve your experience with this class.

CHANGES TO THE SYLLABUS

The instructor reserves the right to change the policies contained in this document at any time. The instructor will notify the enrolled students about any change.