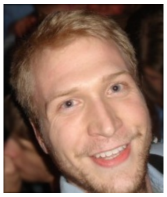
COMP 116: Introduction to Scientific Programming

Class Hours:

Section 1	MWF 12:30 P.M. - 1:20 P.M by Marc Niethammer
Section 2	MWF 9:00 A.M. - 9:50 A.M. by Jeff Pool
Instruction on Mon. and Wed., labs/recitation on Fridays. It is recommended that you bring your laptop to all class sessions.	
Class Room	Section 1: FB009 Section 2: FB007 (Fred Brooks Building: add-on to Sitterson Hall, bottom floor)

Instructors:

Jeff Pool



Office	SN230 (Sitterson Hall, 2nd Floor)	
Office Hours	Mon. 10:00 P.M. - 12:00 Noon, after section 2 (or by appt.)	
Office Phone	976-1760	(Not the best way to get in touch.)
E-mail	jpool@cs.unc.edu	

Marc Niethammer

Office	SN219 (Sitterson Hall, 2nd Floor)	
Office Hours	Wed. 11:00 A.M - 12:00 Noon, before section 1	
Office Phone	843-7449	
E-mail	mn@cs.unc.edu	

TA ???

Office	???	
Office Hours	??? Or by appointment	
Office Phone	???	
E-mail	???@cs.unc.edu	

Catalog Description: **COMP 116** Introduction to Scientific Programming (3).
An introduction to programming for computationally oriented scientists. Fundamental programming skills, using MATLAB and another imperative programming language (such as C). Problem analysis and algorithm design, with examples drawn from simple numerical and discrete problems. *Students can only receive credit for one of COMP 110, **COMP 116**, or COMP 121.*

Prerequisites: Please contact us if you are concerned about whether you have the background required for this course.

MATH 231: We assume familiarity with univariate differential and integral calculus, and the ability to manually solve a system of simultaneous linear equations.

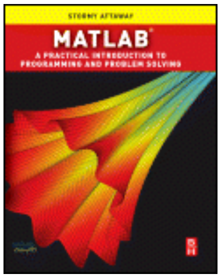
Computer Literacy: Basic proficiency with using a personal computer, using a mouse and keyboard, word processing, email, and finding information off the internet is assumed.

Laptop: Each student needs their own laptop computer for installing and using the MATLAB software during the course of the semester.

Postrequisites: **COMP 116** satisfies the *General Education, Quantitative Reasoning* requirement of 3 credit hours. For students interested in computer science as a major or minor, most upper level and graduate classes in computer science, assume that the student has learned basic programming by taking one of the following courses: COMP 110, **COMP 116**, or COMP 121.

Disabilities: If you have a physical or learning disability, please let us know about your special requirements. For more information, see the following website: http://disabilityservices.unc.edu/

Text Book: *MATLAB: A Practical Introduction to Programming and Problem Solving,* by Stormy Attaway, Publisher: Elsevier, Inc, 2009.



Web Site: We use the Sakai web site, http://sakai.unc.edu, for access to handouts, email and discussion boards. You need your ONYEN to log into Sakai. (See http://onyen.unc.edu if you do not have an onyen. Forgotten passwords can be changed at the IT Response center in the basement of the House undergrad library, make sure to bring your student ID.)

MATLAB software: We will help students install MATLAB on their laptops, during the first Friday session. UNC has a site license that makes MATLAB available to all interested UNC students.
Installation Requirements:

- A Laptop to install MATLAB software on with a working CD-Rom reader (internal or external)
- 700 MB - 2 GB free disk space on your laptop's hard drive.
- Internet access from your laptop
- A valid Onyen and password to login to the UNC MATLAB installation website
- A valid UNC email address <myname@mail.unc.edu> and password to register your academic MathWorks account.
- a valid MATLAB Media CD, we will hand these out during our class on Friday (Aug. 26th).

Course Objectives: After taking this class, students will learn to do…

Numerical Computation: Use MATLAB for doing numerical computation: including arithmetic, algebra, calculus, working with Matrices, and solving systems of linear equations.
Programming: Create programs to solve scientific problems using MATLAB.
Fundamentals: Learn the fundamentals common to many programming languages (variables, data types, flow of control, modular design, etc.).
Debugging: Avoid and track down bugs using defensive programming techniques
I/O: Work with user input (or file input) and transform it into 2D & 3D graphics (or file output).

Invitation: Computers are powerful tools because of their flexibility. They can be programmed to deal with numerous tasks. Once people have learned how to write one program, they can, often with a little more programming, effectively use their area of expertise to best effect. For many modern programming languages, including Java, C++, Python, Tcl/Tk, and Visual Basic, a key principle is to be able to write small fragments of code that interact with sophisticated tools that others have written.

This course introduces basics of programming that are common to many languages. Lectures and assignments draw examples from many scientific disciplines and focus on tasks like reading and analyzing data files and interacting with sophisticated analysis tools.

We focus on the programming language intrinsic to MATLAB, a common tool in mathematics and the sciences. MATLAB makes it easy to work with matrices, graphs, images, and sophisticated mathematical functions. This will allow us to do sophisticated computation (on images, protein molecules, terrain models, etc.) in realistic scenarios as we develop the fundamentals of programming.

Learning a programming language is like learning a spoken language -- *this has several implications:*

Hands-on learning: In our experience, texts and lectures alone are insufficient, one must spend time using a language to make it one's own. The weekly quizzes are to encourage practice in MATLAB.

Journal: Keep a record of things you learn in the lab, lecture, and recitation sections. Whether this is a formal lab notebook, a small diary, or jotted down notes on a word processor. Find a way that works for you to record your insights and ideas.

Reading: Students are strongly encouraged to keep current by reading and reviewing the assigned chapters, handouts, and in-class presentations. This is good preparation for the weekly quizzes, mid-terms, and the final. There is a lot to cover and learn. Cramming at the end of the semester is an unsuccessful strategy for this class.

Elegance: There is always more than one way to say something, but some ways well be more "elegant" than others. You will learn to recognize elegant expressions as you become more familiar with a language and to use them as you become more skilled.

Seeking Help: Language classes are cumulative; don't fall behind. Ask for assistance from your fellow students, the TA's, or from the Instructors, especially if you find yourself struggling.

Attendance: Attendance is recommended in order to stay on top of the material. Lectures and Exams will start promptly at the beginning of each class, Quizzes will start promptly 10 minutes before the end of each class. If you must be absent for an extended period of time due to reasons like an extended illness, please contact us to make arrangements to receive in-class handouts and assignments.

Grading:

Your course grade is based on your cumulative performance on programming assignments, team project, quizzes, and exams as follows:

Quizzes:	20%	*Assignments:*	30%
Midterms:	25%	*Final Exam:*	25%

Quizzes *(10 x 2% = 20%)*: There will be a 10 minute quiz at the end of almost every Wednesday class for a total of eleven quizzes taken. No makeup is given for missed quizzes, but the lowest quiz score will be dropped at the end of the term for a total of ten graded quizzes. The quizzes occur regularly to encourage students to stay up to date on learning the material.

Assignments *(6 x 5% = 30%)*: There are a total of 7 programming assignments. Typically every other week there will be an assignment to write a program. Doing the assignments is essential for learning how to program. Your lowest scoring assignment will be dropped for a total of 6 scored assignments. No makeup is given for missed assignments. Assignments must be submitted before midnight on their due date, which is typically on a Wednesday. Assignments need to be turned in as Microsoft Word documents, both a hard copy (preferably printed out and handed in during the Wednesday class but you can slide it under the TA's door before midnight as well) and a soft copy (submitted via Sakai by the midnight deadline). Once grading has begun, late assignments will not be accepted and will receive a score of zero. Programs that do not work will have 50% of the total assignment points deducted. Remember that programs communicate with people, not just the computer, so write readable code, use brief comments to clarify your intentions and use concise write-ups to explain how you solved the assignment.

Exams *(2 mid-terms x 12.5% = 25%, Final = 25%)*: There will be two mid-terms in class and one comprehensive final exam. These are designed so that you do not need to use MATLAB or perform numerical calculations during the exams. So computers, programmable calculators, and cell phones will not be permitted. You may bring a simple calculator if it gives you a sense of security. If the campus is closed for any reason during a scheduled exam, then the exam will be given during the next scheduled class instead. The final exam will not be given back to students at the end of the semester.

Academic Honesty:

Refer to UNC's honor code at http://instrument.unc.edu/ for a discussion on Academic Honesty. If you are in doubt about any aspect of the honor code, please ask us.

***Collaboration**:* Collaboration on assignments IS encouraged. However, what you hand in must be your own writing / typing. Good scholarship requires that all collaboration must be acknowledged. Thus, if you collaborate on the solution of a problem set, we expect that you list your collaborators at the top of the page. Collaboration on in-class evaluations (quizzes, midterms, the final exam) is, of course, a violation of the Honor Code, see http://www.cs.unc.edu/Admin/Courses/HonorCode.html. This includes discussion of questions on a quiz, midterm, or final with students from a section that have not yet taken that evaluation**.**

Plagiarism: If you consult any outside sources when doing your work, you are expected to fully document those sources. Give credit where credit is due. Plagiarism will not be tolerated.

Cheating: Cheating in any form on assignments, quizzes, or exams will not be tolerated.

Civility:

UNC places a priority on student learning. We value the inherent worth and dignity of every individual. We expect all to be civil to each other by

Etiquette: With the exception of laptops used to take notes or to run MATLAB, please keep all other electronic devices (IE cell-phones) turned off or silenced during class.

Respecting Others: Respect faculty, staff, students, guests, other people's private property, university property, policies, rules and regulations.

Being Responsible: Take responsibility for your choices and actions. Accept consequences for your inappropriate choices and actions.

Being Professional: Communicate in a professional and courteous manner in all forms, and at all times, whether verbal, non-verbal or written.

Course Schedule:

DOW	Date	Lesson / Lab / Event	Readings / Handouts / Tests / Quizzes / Notes
Wed	8/24	Intro to Course Intro to MATLAB	**Quiz: Survey**
Fri	8/26	MATLAB Installation Recitation	**Reading:** 1.1 - 1.4 **Assign #1 Handout:** Image Filtering ***Handout:*** Variable Naming
Mon	8/29	Assignment, Variables, and Expressions	*Last Day to late register (with Late fees)* ***Handout:*** Scripts, Workspaces, Functions

Wed	8/31	Plotting & Publishing Intro. to Matrices & Vectors	**Quiz #1:** MATLAB as a calculator **Assign #2 Handout:** Water Statistics (plotting)
Fri	9/2	Recitation **Help on Assign #1**	**Reading:** Ch. 1.5, 2.1-2.6 ***Handouts:*** MATLAB Reference
Mon	9/5	**Labor Day (Holiday)**	**<No Class>** *9/6 Last day to drop classes (with tuition adjustment)*
Wed	9/7	Matrices & Indices	**Quiz #2:** Vectors & Plotting
Fri	9/9	Recitation **Help on Assign #2**	**Assign #1 Due:** Image Filtering **Reading:** Ch. 1.5 & 11.1 ***Handout:*** Coding Guidelines
Mon	9/12	Matrices, Computations, & Linear Systems	
Wed	9/14	Least Squares & Linear Systems	**Quiz #3:** Matrix Indexing **Assign #3 Handout:** Solving Linear Systems
Fri	9/16	Recitation **Help on Assign #2**	**Assign #2 Due:** Drought Statistics (plotting) **Reading:** Ch. 11.1, 11.2
Mon	9/19	Conditional Logic	
Wed	9/21	Input / Output	**Quiz #4:** Matrices
Fri	9/23	Recitation **Help on Assign #3**	**Reading:** Ch. 2 & Ch. 3 Review for 1st midterm
Mon	9/26	Midterm #1 Review	
Wed	9/28	**1st Mid-term**	**1st Mid-term** **(material covered from 8/24 - 9/16 inclusive)**
Fri	9/30	Recitation **Results for 1st Midterm** **Help on Assign #3**	**Assign #3 Due:** Solving Linear Systems **Reading:** Review Ch. 11.1 & 11.2 **Assign #4 Handout:** Binary Image
Mon	10/3	Functions	
Wed	10/5	Function Challenges	**Quiz #5:** Conditional Logic
Fri	10/7	Recitation **Help on Lab Assignment #4**	**Reading:** Ch. 2.7, 5.1 & 5.2 ***Handout:*** Error handling & Debugging
Mon	10/10	Floating Point Robustness	**Quiz #6:** Functions **Assign #5 Handout:** Terrain Modeling
Wed	10/12	**University Day**	**<No Class>**
Fri	10/14	Recitation **Help on Assignment #4**	**Assign #4 Due:** Binary Image **Reading:** Ch 4.1-4.3
Mon	10/17	Loops (For & While)	*Last day for undergrads to drop classes (with cost)*
Wed	10/19	Common Idioms	**Quiz #7:** Functions & Floating Point **Fall Break begins at 5:00 PM**
Fri	10/21	**Fall Break**	**Reading:** Ch. 4.4 - 4.5 **<No Classes>**
Mon	10/24	Strings Text Manipulation	
Wed	10/26	File I/O & Debugging	**Quiz #8:** 'while' & 'for' logic **Assign #6 Handout:** Protein Data Bank
Fri	10/28	Review **Help on Lab Assignment #5**	**Assign #5 Due:** Terrain Modeling **Reading:** Ch. 6, Ch. 8

Mon	10/31	Cell Arrays & Structures	
Wed	11/2	Pointers, references, & memory storage (in other languages)	**Quiz #9:** Strings & File I/O
Fri	11/4	Recitation **Help on Lab Assignment #6**	**Reading:** Ch. 7 Review for 2nd Midterm
Mon	11/7	Review for 2nd Midterm	
Wed	11/9	**2nd Mid-term**	**(Material covered from 9/19 - 11/4 inclusive)**
Fri	11/11	Recitation **Results for 2nd Midterm** **Help on Assign #6**	**Assign #6 Due:** Protein Data Bank **Assign #7 Handout:** Simulation
Mon	11/14	Advanced Functions	
Wed	11/16	Intro to Diff. Equations	**Quiz #10:** Cell Arrays & Structures
Fri	11/18	Recitation **Help on Assign #7**	**Reading:** Ch. 9
Mon	11/21	More Differential Equations	
Wed	11/23	**Thanksgiving Recess**	**<No classes>**
Fri	11/25	**Thanksgiving Recess**	**<No Classes>**
Mon	11/28	Interpolation Numerical Integration Numerical Differentiation	
Wed	11/30	3D Plotting Intro. to GUI Programming	**Quiz #11:** Differential Equations
Fri	12/2	Recitation **Help on Lab Assignment #7**	***Handout:*** Defensive Programming
Mon	12/5	Visualization	
Wed	12/7	**Review for Final Exam** 1-3, 4.3 - 4.4, 5 - 7, 10.4	**Assign #7 Due:** Simulation **Last Day of class**
		Reading Days	**<No classes> (12/10-12/15)**
Fri	12/9	***Section 1:*** **Comprehensive Final Exam** **(4:00 PM - 7:00 PM)**	***Section 1:*** **Comprehensive Final Exam** **(4:00 PM - 7:00 PM)**
Fri	12/16	***Section 2:*** **Comprehensive Final Exam** **(8:00 AM - 11:00 AM)**	***Section 2:*** **Comprehensive Final Exam** **(8:00 AM - 11:00 AM)**
Mon	12/19	**Final Grades Posted**	

Exam Schedule from

http://registrar.unc.edu/AcademicCalendar/CCM3_032095