CS 151, section 10
Computer Programming Fundamentals for Biologists
Fall 2007

Course Instructor: Melanie Moses
Lecture 1:00-1:50 Mondays and Wednesdays, Mitchell Hall 110
Lab 9:00-11:00 Wednesdays, ESC Pod room 110

Email melaniem@cs.unm.edu to enroll in this section.
The course materials and assignment submission are on WebCT (vista.unm.edu).

Course Objectives
This course will introduce students to computer programming and problem solving using Matlab. It is special section of the introductory course for students not majoring in computer science. Students will learn about computation using familiar biological problems and learn about biology using computational tools. Programming assignments will include bioinformatics, ecological modeling, and data manipulation and presentation. There are no math or programming prerequisites, but students are expected to have completed the biology core (BIOL 201, 202 and 203 and 204). Students who have not completed the biology core may be admitted with instructor permission. This is a 'hands-on' course, so active lab participation is required, and the course grade will be largely based on progressively larger programming assignments. Some programs and lab work will be done in pairs.

By the end of the semester students are expected to

- Be able to write simple computer programs in Matlab
- Apply programming skills to solve biological problems
- Understand basic concepts in computer science such as boolean logic and problem decomposition
- Learn data structures (such as strings, matrices and arrays), logic and control structures (such as 'if' and 'for' statements), data manipulation and presentation (loading data files, computing simple statistics and graphing data), and proper programming techniques (writing modular, well-commented code)
- Understand programming concepts that will assist in learning other languages (like Java, perl, or python)
- Learn about computing resources on campus and online
- Be prepared to enter a second semester programming course (CS 251)

Course philosophy
Many novices find it difficult to program. We will start by writing very small programs to solve simple biological problems. There will be ample time in lab and help sessions for assistance from the instructor and TA's. Once you've gotten comfortable with simple programming, you'll learn more abstract concepts that will help you to write more powerful and interesting programs. The class is lab-focused, so students will spend much more time doing hands-on exercises and writing code than listening to lectures. A few of

the lecture hours will be spent learning about computing resources (for example, a tour of the UNM Arts lab and an introduction to online computing tools.)

Grading
Your grade will consist of programming assignments, lab participation, and tests. Lab participation is essential, and some programming assignments will be started in the lab. Programs will require a significant amount of work outside of class time, so programs will be a large part of your grade. Quizzes during lecture or lab are primarily intended to help you recognize what material is important and where you need further study. Some quizzes will be unannounced.

9 programming assignments: 45%
    The lowest program grade will be dropped.
    Most programs will have a challenging extra-credit option.
Lab participation, exercises and quizzes: 20%
1 midterm: 10%
1 final: 15%
1 final programming project: 10%

Late policy
Programs must be turned in by the due date and time. There is a 10% penalty for each late day. No programs will be accepted after the program solutions have been discussed in class (sometimes this may be after 1 day). Exceptions will only be made for true emergencies with pre-approval from instructor.

Collaboration, online help and academic honesty
Programming is often a collaborative endeavor. Students will work in pairs on some assignments, and are strongly encouraged to use the Web CT discussion boards, to ask the instructor for help, to seek help from computing resources on campus, and to use online help resources.

Students are responsible for documenting all Matlab code that they did not write themselves. The contributions of each student in pair programming assignments must be documented in program comments as shown in class. Any code that was obtained online and modified, must be cited. The original code and web address must be provided in the program comments when it is turned in. Students are encouraged to help each other with concepts from class, but they are not allowed to copy any part of another student's code. In other words, outside of pair-programming assignments, students may help each other by communicating in English, but not in Matlab code.

Any student caught copying code from any source and presenting it as their own will be failed and reported to the University for cheating. Any student who is unclear whether something is 'cheating' should ask the instructor.

Tentative Course Schedule

Week	Lecture Topics	Lab Exercises	Programming Assignment
8/20	Thinking like a computer scientist; computing like an organism	Introduction to the Matlab environment: windows, command lines, saving files	1) Get Matlab up and running, practice entering commands & saving results
8/27	Basic matlab commands, built in functions and variable types	Practice basic commands, scripts & debugging	2) Your first program
9/3	Data types (Arrays, strings & Matrices)	Loading and saving files; manipulating strings and matrices	3) Manipulating matrices: micro-arrays
9/10	Graphing and interpreting data; conditional statements	Graphics Lab	4) Load, manipulate and graph data
9/17	Control structures	Algorithmic thinking & automating repetitive tasks	5) Make a protein; Transcription & translation program
9/24	Problem decomposition; Writing your own functions	Functions	6) An ecological model
10/1	Review	Practice problems to prep for midterm	
10/8	Midterm (10 points);	More advanced control flow & data types	
10/15	Introduction to binary math ArtsLab tour	More graphics & statistics	7) Bioinformatics & string manipulation
10/22	Midterm review; Symbolic equations	Symbolics lab	8) Symbolics & simulation: Isotopes
10/29	Dynamical models & biological simulation	Simulation lab	
11/5	Networks in biology: simple graph theory with applications	Build a simple network	9) Program choice: Bioinformatics or foodweb networks
11/12	Synthesis; revisit computation in biology	Putting it all together	Final project
11/19	How the Internet works; Computing online	Work on final project	Final project
11/26	Advanced topics	Work on final project	Final project
12/03	Review	FINAL EXAM	Final project due