Math 3054 - Fall 2014
Programming and Mathematical Problem Solving

Instructor:  Prof. Christopher Beattie
Email:  beattie@vt.edu
Office:  McBryde 552
Office Hours:  Tues & Wed 11:00am-noon, and by appointment
Phone:  540-231-8279

Lectures:  TTh 9:30am-10:45am, Johnston Student Center (GBJ) 100

Course Objectives:
This course serves as an introductory programming course for Mathematics majors. We will study basic programming techniques for solving problems typically encountered by mathematicians. We start with procedural programming and towards the end of the course move to object-oriented programming techniques. The main programming language for the course is MATLAB.

Textbook and Software:  Insight Through Computing: A MATLAB Introduction to Computational Science and Engineering, Charles Van Loan and K.-Y. Daisy Fan, SIAM, 2010. Additional resources can be found on the book’s website:
http://www.cs.cornell.edu/insight/

Students are required to have access to MATLAB in order to perform computations, edit, and run programs. MATLAB is available for $40 (+tax) to all Virginia Tech students through the Virginia Tech Computing Center. You may download it at http://www2.ita.vt.edu/software/student/. For those who do not wish to install this program on their computers, MATLAB is available on all computers in the Math Emporium, where it can be used without charge.

Prerequisite:  Math 2214

Corequisite:  Math 2224
Syllabus:

Basic procedural programming, including simple programming examples, numerical and non-numerical but focusing on mathematical problems, to demonstrate concepts,
Brief overview of computer architecture focused on what is needed for effective programming and debugging, how computers store programs and data, how do computers execute programs,
Types, declarations, expressions and operators, control flow,
Functions (subroutines) and program structure,
Data structures,
Important theoretical programming concepts (structured programming, top-down design, step-wise refinement).
Graphics and visualization,
Write programs for a number of numerical and non-numerical problems using the procedural approach, from simple programs to programs requiring multiple subroutines and complex data structures, and using graphics/visualization.
Object-oriented programming,
  o Main concepts, abstraction, encapsulation (information hiding), modularity, hierarchy, classes (types),
  o Classes, objects, and functions – implementation in MATLAB,
Write programs for a number of numerical and non-numerical problems using object-oriented approaches.

Grading: The grade will be determined by homework and projects. Homework assignments during the semester will make up 70% of the grade. The final project will take the place of the final exam and is worth 30% of the grade. Homework will be assigned weekly or biweekly. All grades will be posted on the course Scholar site.

An average grade of 90% or better will guarantee an A-, 80% at least a B-, 70% at least a C-, and 60% at least a D.

Assignments: All assignments will be posted on the course Scholar page. The time and date that the assignment is due will be posted with the assignment. Assignments must be submitted electronically through the Scholar site. Late assignments are not accepted unless extenuating circumstances apply (documentation may be required). In case of anticipated special circumstances, you may request extra time, in advance, but extra time may or may not be granted. Two students (i.e., a team of two) may turn in a joint homework. In this case, both students will receive the same grade for that homework. The names of both students must appear on the homework assignment.
Electronic Submission of Programs: Your assignments will require the electronic submission of programs, possibly with output and additional explanations through the class Scholar page. You must follow the following conventions:

1. Program names must contain the assignment number, for example, P343.m, P343_BubbleSort.m.
2. At the start of the program you must include a comment header that states the purpose of the program, (2) states your full name (the author of the program), and (3) the assignment number.
3. Programs and functions must be written in such a way that, when downloaded all into a single directory, the program runs as intended without any additional effort. In particular, this requires that you include all supplemental routines (m-files), including those that were provided to you for the assignment. If you are required to test the program for a number of inputs and/or parameters, you must provide the input used and what output was generated for each input. This can often be done using a script that carries out these tests and/or using MATLAB’s diary function. In many cases you are asked to provide an explanation of observations. This must be included with the assignment. If it may not be obvious how to run your program or function, you must provide an additional file called ‘Readme’ that explains how to run the program or function, and you must provide a script (and input files if necessary) that runs the program or function as intended (providing example input, parameter values, etc.).

In general, your programs will not be accepted unless you have conformed to these rules and conventions, and you will lose points if any or some rules are not satisfied. In particular, programs that terminate with an error condition and no output will be scored with zero points.

Documentation and commenting of programs is crucial to transparency and reproducibility of results. Moreover, good documentation will help you understand what you are doing. All programs must be clearly documented and well commented.

Special Needs: Any student with special needs or disabilities should meet with me as soon as possible if special accommodations are needed to complete the requirements of this class. Alternatively, the student may contact the Dean of Students, who with me, will see that whatever accommodations that are necessary are considered and appropriately addressed.

Honor System: Any write-up and code submitted for grading must be your work alone or joint work with one other student whose name is explicitly listed. The term project must be your work alone. Giving or receiving unauthorized aid or assistance is a violation of the Virginia Tech Honor Code.

Attendance: Please arrive on time and remain until the class is finished.