## MTH 337: Introduction to Scientific and Mathematical Computing. Fall 2015.

**Instructor:** Adam Cunningham
Math Building
adamcunn@buffalo.edu

**Office hours:** MW 4:00-6:00pm

**Lecture 000:** MWF 12:00-12:50pm Baldy 8B
**Lab:** W 11:00-11:50am Baldy 8B

**Lecture 100:** MWF 10:00-10:50am Baldy 8B
**Lab:** F 11:00-11:50am Baldy 8B

### Course Details

**Course Description:** This course provides a broad introduction to computing in the sciences and in both abstract and applied mathematics. It is accessible to students early in their undergraduate program, thereby opening the door to the profitable use of computation throughout the junior and senior years.

The course covers the following areas:

- Programming using Python, the scientific computing package NumPy, and the plotting library Matplotlib.

- Applications of scientific computing in number theory, random number generation, initial value problems, dynamical systems, root-finding, linear regression and optimization.

- Using computers to explore topics in the mathematical and natural sciences.

- Presentation of experiments, observations and conclusions in the form of written reports.

**Prerequisities:** MTH 141 and MTH 142 or MTH 154. No prior programming experience is needed.

### Course Materials

**Textbook:** There is no required textbook for this class. We will be using online resources and extensive class notes. If you would like a book for reference, then either "Scientific Computation" by Bruce E. Shapiro (Sherwood Forest Books, 2015) or "Python for Scientists" by John M. Stewart (Cambridge University Press, 2014) is recommended.

**Python:** We will be using Python 2.7. It is recommended that you use the Anaconda distribution, which is available free on Windows, Mac and Linux and contains all the packages we need (NumPy, SciPy, IPython, Matplotlib).

**LibreOffice:** Weekly reports must be written in LibreOffice Writer, available as part of the LibreOffice suite, which is also free on Windows, Mac and Linux.

**UBlearns:** The course page on UBlearns will be used for posting and submitting assignments.

**Website:** http://www.acsu.buffalo.edu/~adamcunn/MTH337.html

### Coursework

**Quizzes:** Quizzes will be held during recitation approximately every week.

**Reports:** Weekly reports will be submitted via UBLearns. The submission deadline will usually be 9am on a Monday. No late homework will be accepted. In the computation of your overall grade, your lowest homework score will be dropped. You are free to discuss the homework with others, but the work you turn in should be written by you alone.

**Report Portfolio:** You will generate a final portfolio collating all your reports into a single document.

## Grades

**Course Grades:** will be determined by averaging the grades with the weightings:

| | |
|---|---|
| Weekly reports and final portfolio | 70% |
| Quizzes | 20% |
| Class participation | 10% |

**4-point Grading Scale:** Reports will be assigned a letter grade on a four point grading scale:

| | | |
|---|---|---|
| A+ = 4.33 | A = 4.00 | A- = 3.67 |
| B+ = 3.33 | B = 3.00 | B- = 2.67 |
| C+ = 2.33 | C = 2.00 | C- = 1.67 |
| D+ = 1.33 | D = 1.00 | D- = 0.67 |

+/- grades will be used in assigning course grades. Note that the university does not permit A+ or D- grades.

**Academic Honesty:** Students are expected to adhere to the university policy on academic honesty. Cheating, plagiarism, or misrepresentation of your work will result in formal charges.

**Incompletes:** Incompletes will be given only under extraordinary circumstances (like surgery during the last week of class).

## Expectations for this class

**Class participation:** Aspects include attending class, asking and answering questions in class, contributions to any group written documents, and involvement in other group activities.

**Independent research:** We cannot cover every single option available in Python in the limited time we have in class. It is expected therefore that you make use of the online resources to do research for youself. If there is something that you want to do, the chances are that Python already has a way to do it - you just need to find it.

**Computer use: Absolutely no non-class-related computer or phone use during class** (e.g. texting, Facebook, emailing): the penalty will be one third of a letter grade off overall course grade for each instance.

**How to succeed:** Sustained steady effort, starting from day one. Make full use of the resources available to you: the classnotes, website, online documentation, and each other. See me in office hours, email me, or make an appointment to see me at some other time. Start homework assignments early - programming and debugging often take more time than anticipated.

## Other

**Important dates:**

Tue Sep 8  Last day to drop the course - no record appears on transcript.

Fri Nov 13  Last day to resign from the course - an R appears on the transcript.

**Students with Disabilities:** If you have a diagnosed disability (physical, learning or psychological) which will make it difficult for you to carry out the course work as outlined, or requires accommodations such as recruiting note takers, readers or extended time on exams and/or assignments, please advise me during the first two weeks of the course so that we may review possible arrangements for reasonable accommodations.

## Tentative Class Schedule

| Week | Date | Programming | Topics |
|------|------|-------------|--------|
| 1 | 8/31 | Introduction to IPython notebook<br>Python numbers, variables, lists | The Euclidean algorithm |
| 2 | 9/7 | Python loops, functions, namespaces<br>Matplotlib basic plotting | Number Theory: Exploration in the integers |
| 3 | 9/14 | Python string formatting<br>NumPy arrays and vectorization | Prime number generation<br>Deterministic dynamical systems |
| 4 | 9/21 | Python list comprehensions<br>NumPy array indexing and slicing | Floating point numbers<br>Floating point arithmetic |
| 5 | 9/28 | Loading data from files | Hazards of floating point arithmetic |
| 6 | 10/5 | Matplotlib contour plots | Linear least squares<br>Solving differential equations |
| 7 | 10/12 | Matplotlib animation<br>NumPy fancy indexing | Initial value problems<br>Root finding methods |
| 8 | 10/19 | RGB color representation<br>Image creation and display | Convergence of root-finding methods |
| 9 | 10/26 | Matplotlib histograms | Random number generators<br>Monte Carlo integration |
| 10 | 11/2 | NumPy random number generation | Non-uniform random numbers |
| 11 | 11/9 | Matplotlib drawing | Minimizing functions of one variable<br>Minimizing multivariable functions |
| 12 | 11/16 | PuLP linear programming modeler | Linear programming |
| 13 | 11/23 | Timing code execution | Thanksgiving |
| 14 | 11/30 | Recording and loading WAVE files | Data acquisition and analysis |
| 15 | 12/7 | Python dictionaries<br>String manipulation functions | Computing with text |