Introductory Computer Programming for METOC (MR2020; Autumn Quarter 2021)
Officially Computer Computations for Air-Ocean Sciences
Instructor: Scott Powell (Root 255)
Room: IDEA Lab (Root 123)
Meeting times: T,R: 1100–1200; F: 1000–1200
Course webpage: http://faculty.nps.edu/scott.powell/MR2020.html

Course Objectives

- Learn to use and navigate UNIX based systems (e.g., Mac, Linux).
- Learn fundamentals of using computer logic to translate an outline of desired operations into computer code
- Familiarization with developing code in the python programming language to accomplish common tasks such as and reading/writing data, executing computations on that data, and plotting one-dimensional and multi-dimensional data
- Build your own code from scratch to read in data and generate your own plots in the form of a final project.

Syllabus

The syllabus is approximate. We will plan to cover all material listed, but the timing may differ from the schedule depending on the median learning pace of the class. In addition, there may be material covered that is not listed on the syllabus depending on the needs of students and on questions raised during class.

Week 1 (Sept. 27–Oct. 1): Using a UNIX terminal. Accessing Jupyter Notebooks via online JupyterHub. Using JupyterHub. Opening python notebooks. Running a simple code. Module imports. Searching for documentation and new modules.

Week 2 (Oct. 4–8): Python data types and variables. Operations. Comparisons. Introduction to pseudo-coding. Leaving comments in code.

Week 3 (Oct. 11–15): If-else blocks. For and while loops. Try-except blocks.

Week 4 (Oct. 18–22): Multi-dimensional arrays; Indexing; Calling functions within modules (emphasis on numpy); some important functions (e.g., [nan]mean, [nan]median)

Week 5 (Oct. 25–29): Functions and classes; passing variables between functions; importing user-made functions

Week 6 (Nov. 1–5): Using python containers (lists and dictionaries). List comprehension; Introduction to pandas.

Midterm Exam: November 5 (1000–1200) **Material through Week 5**

Week 7 (Nov. 8–12): Data I/O (Text, HDF, NetCDF, grib, pickle files, numpy files, MATLAB .mat files)

Week 8 (Nov. 15–19): Continue Data I/O

Week 9 (Nov. 22–24): *No class Thursday or Friday (Thanksgiving).* Matplotlib: Create figures and subplots, making line plots, labeling axes, creating legends

Week 10 (Nov. 29–Dec. 3): Matplotlib: Two-dimensional plots; color palettes and bars.

Weeks 11 and 12 (Dec. 6–17): Final coding project. Classroom will be open during regular meeting hours.

Grading

Midterm (30%)
Final project (70%)
No extra credit will be offered.

Approximate grading scale (after highest grade is curved to 100*):
These are the minimum letter grades students can receive if their numerical grade is within the range shown in the left column. For example, a student with a final grade of 87 will have earned no lower than a B+.

Numerical Grade	**Minimum Possible Letter Grade**
93–100	A
90–92	A-
87–89	B+
83–86	B
80–82	B-
77–79	C+
73–76	C
70–72	C-
67–69	D+
63–66	D
60–62	D-
Below 60	X/F

*The difference between 100 and the highest student grade at the end of the course will be added to each student's grade before being converted to a letter grade. For example, if the highest course grade is 93, 7 points will be added to each student's grade. The new total with the additional 7 points will then be converted to a letter grade.

Course Structure:

1. Course material will be available at or linked from the course webpage.

2. The Teams chat group for MR2020 is the venue for group discussion and questions. Students may also reach the instructor via private chat on Teams or email.

3. Live meetings for this class will be held in-person and broadcasted live on Teams unless circumstances change during the quarter. Recordings will be saved if possible, although students may not see drawings on the board. An effort will be made to combine use OneNote as a virtual "board" that is also displayed in the classroom. However, remote students will likely not see some of the material presented on the classroom whiteboards.

4. Students on Teams should actively participate in discussion using audio and chat as appropriate or if called upon.

5. Due to limited Navy financial support for NPS instruction, the instructor cannot be available for either virtual or in-person meetings outside of class. The instructor will only respond to emails or messages on Teams related to class infrequently. If at all possible, questions should be raised in class.

6. The final project will be graded based on the following:

- Did the student generate the correct final product/output and can the code be run by the instructor? (50%)
- Is the code readable, properly structured (i.e., proper use of functions, etc.) and well-commented? (30%)
- Is the code efficient (i.e., run time and memory usage minimized)? (20%)