Introduction to:
Computers & Programming: Administrative Matters

Adam Meyers
New York University
Outline

• What to Expect from a Intro to CS Class
• Basic Info:
  – Class Schedule, Room Number, Office Hours, Contact Info, Course Webpage
• Policies:
  – Homework, Exams, Grading, Cheating
• What is covered in the Class
  – Materials, Syllabus, Structure of classes
• Installing Python
What to Expect from Intro to CS

• You will learn how to write computer programs:
  – Sets of instructions computers follow to solve problems, display images, etc.
  – The end result of creative problem solving

• Skills that are helpful:
  – Ability to follow instructions loosely
  – Experimentation and game play when solving problems
  – Combination of mathematics (especially logic) and creativity
  – Asking questions without fear
  – Willingness to fix things when they don't work (and not being afraid of breaking something).

• My best students varied a lot in their specialties: mathematicians, artists, filmmakers, philosophers, musicians, etc. …
What to Watch Out For

- Computer Science is a type of Math: info is sequential
- Key concepts can hold you back if you don't understand
  - Similar to disciplines including: art, music, math, …
- May be more difficult than a typical “Intro to” Elective
- Do not start this class late in the term
- Do not hesitate to ask questions in class
- Do not do homework late or miss classes (lectures or labs)
  - This can have a substantial effect on test grades
  - There is no (official) penalty for missed classes or late HW (until end of the term), but doing so impedes understanding
- If you don't understand, seek help immediately so you don't fall behind
  - Ask tutors (lab classes, computer labs, etutors)
  - Ask me (office hours, appointment, email)
- If you have trouble & are grade conscious, consider taking class Pass/Fail
Basic Info: CSCI-UA.0002 Spring 2018

- **Websites**
  - Section 004: https://cs.nyu.edu/courses/spring18/CSCI-UA.0002-004/
  - Section 009: https://cs.nyu.edu/courses/spring18/CSCI-UA.0002-009/
  - CSCI-UA.0002 Website: https://cs.nyu.edu/courses/spring18/CSCI-UA.0002-003/common_syllabus/
  - Professor's Website: https://nlp.cs.nyu.edu/people/meyers.html

- **Schedule:**

<table>
<thead>
<tr>
<th>Section</th>
<th>Days</th>
<th>Time</th>
<th>Room</th>
</tr>
</thead>
<tbody>
<tr>
<td>004</td>
<td>M &amp; W</td>
<td>11:00-12:15AM</td>
<td>60FA 110</td>
</tr>
<tr>
<td>009</td>
<td>M &amp; W</td>
<td>8:00-9:15AM</td>
<td>60FA C10</td>
</tr>
</tbody>
</table>

  - **No Classes:** Mon February 19, Mon March 12 and Wednesday March 14
  - **Midterms 1 and 2:** Wednesday October 11 and Wednesday November 15
  - **Final:** Same Classroom, Different Timing
    - Section 004: Monday May 14, 2018 10:00AM—11:50AM
      - In the past, finals have been in the same rooms as the class (but rooms not listed yet)
    - Section 009: Monday May 14, 2018 8:00AM—9:50AM
  - Please **plan your plane tickets** for some time after the finals are scheduled to occur.
  - Permission to take the test for the other section depend on availability (and legal room capacity)

- **My office:** 60 Fifth Avenue, Rm 301
- **Office Hours:** Monday: 1:30-3PM or Thursday: 10:30-12PM or by appointment
- **Email and Phone:** meyers@cs.nyu.edu and 212-998-3482
Website and NYUClasses

• NYUClasses
  – Submitting Homework
    • Usually python code or text
    • Other HomeWork Should be in pdf form (e.g., scan of drawn flowchart)
  – Getting Grades
  – Course Documents with Licensing Restrictions
  – Quizzes (connected to Modules)

• Website – All Other Course Information
  – Schedule, class lectures, sample test questions, python information, etc.
  – Modules (Supplemental videos & Interactive Learning)
  – Updated throughout the semester (errors fixed, new stuff added)
Intro: Computers & Programming
Administrative Matters

CSCI-UA.0002

Grades

• Online Quizzes 5%
• Homework (programming assignments): 20%
• Midterm 1 20%
• Midterm 2 20%
• Final 35%

• All grades are converted to letter grades before averaging (A = 4, A- = 3.7, B+ 3.3, ..., D = 1, F = 0)

• Interpreting grades without grade Inflation
  – A and A- are good (well above average) grades
  – B+, B and B- are above average, average, and slightly below
  – C+ and C may indicate you are finding the material difficult
  – D means barely passing and F means failing (very rare)
def calc_grade1(Quiz, HW, Mid1, Mid2, Final):
    # Quiz is the average grade for quizzes
    # HW is the average grade for programming assignments
    # Mid1 is your Midterm 1 grade
    # Mid2 is your Midterm 2 grade
    # Final is your grade on the Final Exam
    Grade = (Quiz * .05) + (HW * .2) + (Midterm1 * .20) + (Midterm2 * .20) + (Final * .35)
    return(Grade)

def calc_grade2(Quiz, HW, Mid1, Mid2, Final):
    Grade = (Quiz * .05) + (HW * .2) + (max(Midterm1, Midterm2) * .30) + (Final * .45)
    return(Grade)

def final_grade(Quiz, HW, Mid1, Mid2, Final):
    Grade = max(calc_grade1(Quiz, HW, Mid1, Mid2, Final), calc_grade2(Quiz, HW, Mid1, Mid2, Final))
    return(Grade)

• Bonus for A Grade on 2 Midterms: One A counts as an A+ for purpose of calculating average
• Resulting grade will be rounded up or down to a grade between A and F
  – 2.95 will become a 3 (B) and 2.75 will become a 2.7 (B-)
  – Cutoff between B and B- will depend on grade distribution of class, e.g., the cutoff may be 2.85 or 2.91
Cheating

- Tests, homework and grades are:
  - Tools for assessment of what you did and did not learn
- Cheating is when you copy someone else's work without giving them credit.
- Discussing homework with classmates is not cheating if:
  - Each student produces a different solution and/or
  - Afterwards, each student could solve similar problems on their own
  - For large collaborations, students should cite each other's name on the HW
- Consulting other students on tests is definitely cheating.
- Please do not cheat:
  - You will fail the class (if you are caught) and be reported to the administration
  - Even if you don't get caught, it will negatively impact learning the material
  - If you are so worried about your grade, that you feel compelled to cheat, take the class Pass/Fail instead. If you take all the tests, you are extremely unlikely to fail the class.
Late Homework, Missing Classes and Grades

• It is Easier to Learn the Material in Class (including labs)
  – Reading online lecture notes is not the same as attending
  – I sometimes present material differently from the modules and from the book
    • For example, I place greater emphasis on writing code using functions
    • This can make relying on online materials more difficult

• Multiple-choice quizzes – no credit if late (but they are not worth very much anyway)

• Programming Assignments: Lateness → Natural Consequences, Fairness and Guilt
  – Natural Consequences
    • Easier to prepare for tests if you have graded homework on the material on the test
    • Doing homework on time makes it easier to keep up because CS is sequential
  – Fairness: The graders mark on-time homework first and then late homeworks
  – Guilt:
    • Graders have lives too – it is an imposition on them when you do late HW
    • They are paid by the hour – if they run out of time, they may not have time to grade your late homework, e.g., they have to pass their classes too
    • Extremely Late HW may never be graded if the grader simply does not have time
  – No Points Taken Off for late programming assignment unless it is the End of the Term and Grader simply does not have time. Once a homework is 3 weeks late or it is the last week of the term, there is no guarantees that late homework will be graded. HW 10, which is optional and due on the last day of class does have a hard deadline.
  – I think these consequences are severe enough without me taking additional points off, which I think would compound the problem.
If You Miss a Class, Please keep Up With the Material

• Check the website
• Pay me an office visit
• Consult with a classmate (get their contact info)
• Plan ahead so you can go to the corresponding class in another section (ask me first, please.)
Homework

- Chapters in the Gaddis Book (and some other reading)
- Online modules
- Online multiple choice quizzes after each module (5%)
- Programming Assignments (20%)
  - Each program is judged as follows:
    • Does it solve the problem posed by the assignment
    • Is it clear?
    • Does it work?
    • Do you go beyond the assignment and show that you really know what you are doing? Do you add “bells and whistles”?
Why Does it Matter That I am Teaching 2 Sections

• It is possible to makeup missed classes and exams
  – It is OK to switch classes occasionally, to make up a test or a missed class, etc. as an exception to the rule
    • provided that you let me know (by email) and one class does not become overcrowded and disfunctional.
      – Overcrowding is illegal due to the fire code
    – In terms of make-up classes, the alignment may be imperfect, e.g., if one class ends up going more quickly than the other for any reason (e.g., fire drills, bad weather, etc.)
  – Shared office hours (with a 3rd class on another topic as well).
Syllabus

• Introduction to Programming Languages
• Python Basics
• Working with variables and operators
• Control Structures
• Repetition Structures
• Working with Text
• Functions and Modules
• Lists and Dictionaries
• File Input and Output
• Other topics, as time permits
Structure of Classes

• **Types of Classes**
  – 40%: lecture class, sometimes with in-class program
  – 33%: lab-style class.
    • May include material I did not have time for during previous lecture
    • Everyone will do their “homework” in class with the support of T.A.s and myself
    • Finish Homework at Home and send in (approximately 1 assignment per week)
  – 27% Reviews and Tests
    • There are 3 tests and 3 pre-test reviews and 2 post-test reviews out of 29 sessions

• **Laptop Recommended, but if you do not have a laptop**
  – We will partner you with someone who does and you will both include each other's names on your homework so the grader knows who you partnered with.
  – You will be able to get copies of your work by email and submit it later.

• **11 Modules (Online materials) and Associated Quizzes:**
  – Modules: Include short videos and testing of code in workspaces
  – Multiple choice quizzes with strict deadlines, worth a small part of the final grade.
Materials

- **Starting with Python (3rd or 4th Edition)** by Tony Gaddis
  - Cover most (not all) material in book, as per HW assignments
  - Rate of reading: 1 chapter every 2 or 3 classes
  - Instructions for adapting the 2nd Edition for use with this class

- **Modules: Self-Study Materials on NYU Website**
  - Links on Class Website
  - Matching Quizzes in NYUClasses

- **Slides and Notes from Lectures**: most material available online, but copyrighted material in NYUClasses (just 1 set of slides)

- **Python Documentation** (reference): [https://docs.python.org/3/](https://docs.python.org/3/)

- **Additional material**: How computers work, algorithms, properties of other programming languages, etc.
Install Python ASAP

- Go to [www.python.org](http://www.python.org)
- Click on the Download tab
- Download Latest Version of **Python 3** (NOT Python 2)
  - Probably Version 3.6
    - Compatibility issues may be solved by installing slightly earlier version (3.5)
  - **Please Install Python 3 for Windows, MAC or Linux**
    - **Other platforms are possible, but I advise against using them**
      - Tablets, cell phones or Chromebooks
      - You could run into extra difficulties and find it difficult to get support
- **Install Python**
  - If there are installation problems, do not be silent
  - For additional Instructions go to this site and click on “View Instructions”
    - [https://cs.nyu.edu/courses/spring18/CSCI-UA.0002-003/common_syllabus/#software](https://cs.nyu.edu/courses/spring18/CSCI-UA.0002-003/common_syllabus/#software)
  - If you have a laptop, take your laptop to the **lab tutors**
    - Computer Lab at Kimmel Hall
    - Schedule will be posted at common curriculum website (may start 2\textsuperscript{nd} Week of Classes):
      - [https://cs.nyu.edu/courses/spring18/CSCI-UA.0002-003/common_syllabus/#tutoring](https://cs.nyu.edu/courses/spring18/CSCI-UA.0002-003/common_syllabus/#tutoring)