CSc 120
Introduction to Computer Programing II

00: Basic info

Basic info about this class

- Second programming class
  - Prerequisite: CSc 110 (or some programming experience)
  - looks at
    - more complex programs and programming problems
    - how data are represented and manipulated
    - how to think about and understand program behavior
- Assumes you have at least a little programming experience
  - can write small programs; execute, test, and debug them
  - **if not: take CSc 110**

Basic info about this class

- Programming language: Python
  - we will use Python 3
  - first few lectures : review basics
- Development environment: idle
  - comes with python
- If you don’t know python:
  - need to pick up the basics quickly
    - let me know
    - work with tutors and SLs

Instructional staff

- Instructor: Janalee O'Bagy, Ph.D.
  - Office: Gould-Simpson room 854
  - Email: jobagy@cs.arizona.edu or jobagy@email.arizona.edu
  - Office hours: MW 10am-12pm
    or if my door is open
    or by appointment (send email; put **CS 120** in Subject:)
- Discussion sections:
  - 50 mins, Tue; led by Section Leaders
  - see class website for details

Course communication

- Piazza
  - http://www.piazza.com
  - Sign up if you haven't already!
  - Questions are posted and answered here
  - See Piazza for updated office hours

Class Website

  - Important links: assignments, email contacts, Python info, etc.
  - https://www2.cs.arizona.edu/classes/cs120/fall17

Textbook

- No required text
  - you will be given the information you need
  - plenty of additional on-line resources available
  - https://docs.python.org/3
- Optional text:

  Problem Solving with Algorithms and Data Structures using Python (2nd ed.), by Bradley Miller and David Ranum.

  Franklin Beedle & Associates, 2011. ISBN 978-1-59028-257-1.

Assignments

- Typically, once a week
  - given out on Wed
    - several small problems:
      - auto-grader
      - due Saturday 7pm
    - one or two larger problems:
      - due following Thursday 7pm
  - graded feedback back to you by following Monday
- ~ 11-13 assignments over the entire semester
  - best 10 scores considered for grade

Assignments

- Due at time specified
  - no late submissions accepted
  - in almost all cases: no extensions

  - plan ahead
    - plan around assignments etc. for other classes
  - start early
    - procrastination is the surest way to sabotage your performance in this class

Assignments

- Grading:
  - coding style
    - code structure, comments, etc.
  - functionality
    - tested using a computer program
    - you need to follow directions *exactly*
      - file names
      - function names
      - input/output format
      - … anything else specified…

Taking liberties with assignment specs is *not* the right way to show your creativity!

Exams

- Two midterms
  - approx. six weeks apart
    - see syllabus, website for dates
    - count for 30% of final grade (2 x 15%)
- Occasional quizzes (not graded)
- Final exam:
  - Wednesday, Dec. 13th, 2017 at 8:00am
    - counts for 25% of final grade

Midterms

- Start at the beginning of lecture period
- About 50 mins each
- No make-up exams except for unforeseeable emergencies

Grading policy

Components of your final grade:

programming assignments (best 10):	40%
weekly section participation:	5%
midterms	30%
final exam:	25%

Regrade request deadlines:

- programs: within two weeks of getting grade back
- within one week of getting grade back

Grading policy

Grade boundaries:

90% and above:	A
80% and above, but below 90%:	B
65% and above, but below 80%:	C
55% and above, but below 65%:	D
Below 55%:	E

(I may lower the cuttoffs but will not raise them.)

Behavior and conduct

- treat each other with respect and courtesy
- don't be disruptive

- these behaviors will not be tolerated in class:
  - phone conversations, texting
  - reading newspapers or magazines
  - games, facebook, other social media
  - extended conversations (unless instructed)

please leave the room if you *have* to do any of these activities; come back when done.

Academic integrity

- Any work submitted for credit must be your own work
  - OK:
    - general discussions of how to approach a problem
  - NOT OK:
    - discussing the specifics of the code for an assignment
    - partnering with someone else on an assignment
    - soliciting help on online forums (e.g., stackoverflow)

Academic integrity

- Helping someone else cheat is just as bad as cheating yourself:
  - don't show your code to anyone else
  - don't share details of code ahead of a submission deadline
  - don't post your assignment code publicly
    - this includes Piazza
    - see me or the SLs to discuss your code

- See syllabus for detailed list of do's and don’t's

How to succeed in this class

- Understand the material
  - if you don't: *ask questions!*
- Attend sections
  - *participate!*
- Do the programming assignments
  - start early (no late days)
  - follow directions *exactly*
  - test your code thoroughly
  - *don't forget to submit your code!*

First section (tomorrow)

- Required attendance
  - Survey on Piazza
  - Due Tuesday, August 22nd at 4:00pm
  - It will take 5 minutes