CSc 120
Introduction to Computer Programming II

00: Basic info
Welcome to CSc 120

• Second programming class
  - prerequisite: CSc 110 (or some programming experience)
  - looks at
    o more complex programs and programming problems
    o how data are represented and manipulated
    o how to think about and understand program behavior
  - start building the toolbox of a computer scientist

• Assumes you have at least a little programming experience
  - can write small programs; execute, test, and debug them
  - if not: take CSc 110
Instructional staff

• Instructor: Janalee O'Bagy, Ph.D.
  – Office: Gould-Simpson room 854
  – Email: jobagy@email.arizona.edu
  – Office hours: MW 1:00 – 2:30pm
    or if my door is open
    or by appointment (send email; put CS 120 in Subject:)
    Check the website for updates to office hours
Instructional staff

• Teaching Assistants (TAs)
  – Jonathon Davis
  – Paria Khamsehzadeh
  – Victor Gomes

• TA office hours:
  – Will be posted on Piazza
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
  o make use of office hours!
Course communication

• Piazza
  – http://www.piazza.com
  – Sign up if you haven't already!
  – Questions are posted and answered here
  – Class communication takes place here

• Class Website
  – Important links: assignments, email contacts, syllabus, etc.
  – https://www2.cs.arizona.edu/classes/cs120/summer18/
  – See Piazza for updated office hours
Textbook

• Text:

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

• Additional resources
  – given as needed
  – plenty of additional on-line resources available
  – https://docs.python.org/3
In-class activities and Section Activities

• In-class activities (ICAs):
  – Activities (problems sets or quizzes) that are graded
  – Work in groups (except for quizzes)

• Sections (for summer session):
  – 50 mins, typically on Friday’s (last portion of class lecture)
  – Section activities are graded for participation, not correctness

• Online students will submit via Gradescope
  – Details to come
Meet your neighbor (2 minutes)

• Find out where they are from

• Together, decide on answers to these questions
  – What year was Python created?
  – How many websites are there today and how many were there 20 years ago?
  – The Python we use is written in C. How many lines of C code do you think it takes to implement Python?
Assignments

• Normal schedule
  – given out on Wed
    o several small problems:
      • auto-grader
      • due Saturday at 11:59pm
    o one or two larger problems:
      • due following Thursday at 11:59pm
  – graded feedback back to you by following Monday
  – Week 1: short problems due Thurs; long problems due Sat

• ~ 9-10 assignments over the entire summer session
  – no assignments are dropped
  – do them all!
Assignments

• Due at time specified
  – 2 late days (long problems only)
    o may submit up to 24-hours late
    o may **not** use late days for the first and last assignments
  – in almost all cases: no extensions

➢ plan ahead
  o plan around assignments for other classes, other commitments

➢ start early
  o procrastination is the surest way to sabotage your performance in this class
Assignments

• Grading:
  – coding style
    o code structure, comments, etc.
  – functionality
    o tested using a computer program
    o 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. 3 weeks apart
    o see syllabus, website for dates
    o count for 30% of final grade (2 x 15%)

• Final exam:
  – Friday, August 8\textsuperscript{th}, 2018 at 10:00am
    o counts for 15% of final grade
Midterms

• In-class students
  – start at the beginning of lecture period
  – regular lecture then follows

• Online students
  – we use Examity for proctoring
  – Online students need a computer with a webcam
  – must be completed by the end of lecture on the exam day

• 50 mins each

• No make-up exams except for unforeseeable emergencies
Grading policy

Components of your final grade:

<table>
<thead>
<tr>
<th>Component</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>programming assignments</td>
<td>45%</td>
</tr>
<tr>
<td>in-class activities</td>
<td>5%</td>
</tr>
<tr>
<td>weekly section participation</td>
<td>5%</td>
</tr>
<tr>
<td>midterms</td>
<td>30%</td>
</tr>
<tr>
<td>final exam</td>
<td>15%</td>
</tr>
</tbody>
</table>

Regrade request deadlines:
- programs: within 1 week of getting grade back
- midterms: within 1 week of getting grade back
## Grading policy

**Grade boundaries:**

<table>
<thead>
<tr>
<th>Percentage</th>
<th>Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>90% and above</td>
<td>A</td>
</tr>
<tr>
<td>80% and above, but below 90%</td>
<td>B</td>
</tr>
<tr>
<td>70% and above, but below 80%</td>
<td>C</td>
</tr>
<tr>
<td>60% and above, but below 70%</td>
<td>D</td>
</tr>
<tr>
<td>Below 65%</td>
<td>E</td>
</tr>
</tbody>
</table>

(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:
    o general discussions of how to approach a problem
  – NOT OK:
    o discussing the specifics of the code for an assignment
    o partnering with someone else on an assignment
    o 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
    o this includes Piazza
    o see me or the TAs 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!*
  – office hours

• Attend lecture and section
  – *participate!*

• Do the programming assignments
  – start early (only 2 late days)
  – follow directions **exactly**
  – test your code thoroughly
  – *don’t forget to submit your code!*
The muddiest point

• What was the thing that you understood the least?
  – Specific to your prereq class (110, ECE 175, etc.)

Write it down!
Survey

• Required
  – Survey on Piazza
  – Due Tuesday, June 4th at 7:00pm, MST
  – It will take 5 minutes

Part of your grade!