Welcome to CSE 123!

Nathan Brunelle
Spring 2024
Agenda

• About us
• About this course
  • Learning objectives
  • Other similar courses
  • Course components
• Our learning model

• Tools and resources
  • Course Website
  • Ed
  • VS Code
• Assessment and grading
• Collaboration
Agenda

• About us

• About this course
  • Learning objectives
  • Other similar courses
  • Course components

• Our learning model

• Tools and resources
  • Course Website
  • Ed
  • VS Code

• Assessment and grading

• Collaboration
Hi, I’m Nathan! (he/him)

• Associate Teaching Professor
• New to UW as of Sept. 2024
• First time teaching 12X
• Interested in CS education/pedagogy
• Previously:
  • Taught at U. Virginia for 6 years
  • Took a cross-country road trip with my spouse and dog
  • Taught CSE 332 (autumn 2023, winter 2024)
Meet (most of) your 30 TAs!
Agenda

• About us
• About this course
  • Learning objectives
  • Other similar courses
  • Course components
• Our learning model

• Tools and resources
  • Course Website
  • Ed
  • VS Code
• Assessment and grading
• Collaboration
Learning Objectives

or, “What will I learn in this class?”

Seven themes:

• Computational Thinking
• Code Comprehension
• Code Writing

• Communication
• Testing
• Debugging
• Ethics/Impact
Prerequisite Knowledge

- Comfort with control structures
  - loops, conditionals, methods/functions

- Experience with using basic data structures
  - arrays, lists, sets, maps

- Experience with console and file input/output

- Exposure to simple object-oriented programming
  - classes, interfaces

- Programming experience *in Java*
  - Or willingness to pick up on your own
## Other Similar Courses

<table>
<thead>
<tr>
<th>Course</th>
<th>Good choice if...</th>
</tr>
</thead>
</table>
| CSE 123 | • You done a fair bit of programming, at least some of which is in Java AND  
|         | • You are, or want to be, in a major such as CS, CE, ECE, Info, etc. that requires Java programming OR  
|         | • You’re interested in creating software (whether as a hobby, side-gig, career, etc.) |
| CSE 122 | • You’ve done some programming (roughly one course worth) in *any* programming language AND  
|         | • You are, or want to be, in a major such as CS, CE, ECE, Info, etc. that requires Java programming |
| CSE 143 | • You took CSE 142 at UW, at a community college, or through UW in the High School |
| CSE 163 | • You’re interested in data science and analysis OR  
|         | • You want to learn Python* OR  
|         | • You are, or want to be, in a major such as Physics, Bio, Stat, etc. where analyzing data through programming is useful |
| CSE 154 | • You’re interested in web development (HTML, CSS, JS) |

See [Guided Self-Placement](#), [Introductory Courses](#), and [CSE 143/143X](#) for more info
Help Us Improve!

• CSE 123 is *very new*!
• We worked hard to build a course we think will be effective and supportive and help you succeed
• We probably didn’t get it all right

• We appreciate your patience and understanding if we need to make adjustments during the quarter
• Please give us lots of feedback!
  • Post on Ed and/or use the Anonymous Feedback Tool
Course Components

Lessons (aka Lectures)
• WF, 10:30 or 2:30
• Held live on campus; recordings released after
• First introductions to course concepts
• Mix of presentation of content and practice activities/problems
• Required (but not graded) pre-work for most sessions

Sections
• TuTh, various times
• Led by TAs
• Held live in person; not recorded
  • Materials will be released online afterwards
• Additional review, discussion, and practice
• Mostly practice problems

Attendance is not taken, but you are responsible for all material (including announcements).
Agenda

• About us
• About this course
  • Learning objectives
  • Other similar courses
  • Course components
• Our learning model

• Tools and resources
  • Course Website
  • Ed
  • VS Code
• Assessment and grading
• Collaboration
Course Culture and Support

• Currently 600 students enrolled!
  • Wide range of backgrounds, interests, and goals
• Support and help each other!
  • Form study groups
  • If you have a question, others almost certainly do too
• Lots of ways to get support from us
  • Message board, IPL, section
Course Culture and Support

• Policies designed with flexibility in mind
  • Resubmissions, ignoring quiz problems, lecture recordings, etc.

• But life and the world still happen...

• Please reach out ASAP if you’re struggling or have circumstances that require extra support
Agenda

• About us
• About this course
  • Learning objectives
  • Other similar courses
  • Course components
• Our learning model

• Tools and resources
  • Course Website
  • Ed
  • VS Code
• Assessment and grading
• Collaboration
Course Website

[cs.uw.edu/123](cs.uw.edu/123)

- Primary source of course information (*not* Canvas)
- Calendar will contain links to (almost) all resources
Please review the syllabus ASAP.
Ed

• Our online learning platform
• Submit graded work
• Receive/View feedback
  • Including announcements
Questions (2:30)

• Can we use break command?
• What kind of help is there for brushing up on java or other CSE122 content?
• Do resubmission replace the entire grade
Questions (10:30)

- Grade calculator?
- How do I determine my grade?
- Final hand written?
- Are all assignments submitted through Ed?
- Practice tests vs section question for course prep?
- IDE?
P0: Warmup/Review

Will be released today or tomorrow, on Ed.

Not the standard format for assignments going forward, intended to be a series of shorter review questions.

Due Wednesday (4/03)
Agenda

• About us
• About this course
  • Learning objectives
  • Other similar courses
  • Course components
• Our learning model

• Tools and resources
  • Course Website
  • Ed
  • VS Code
• Assessment and grading
• Collaboration
Assignments and Grading

• Our goal in the course is for you to gain proficiency the concepts and skills we teach.

• We assess your proficiency by asking you to apply the concepts and skills on tasks or problems.

• By necessity, we are assessing your work as a proxy for your proficiency.
Assignments

• Your learning in this course will be assessed in four ways:
  • Programming Assignments (~biweekly, 4 total)
    • Structured programming assignments to assess your proficiency of programming concepts
  • Creative Projects (~biweekly, 4 total)
    • Smaller, more open-ended assignments to give you space to explore
  • Quizzes (3 total, in section)
    • Series of problems covering all material up to that point
  • Final Exam (Monday, June 3)
    • Final, culminating assessment of all your skills and knowledge
Resubmission and Quiz Problem Drops

Learning takes time, and doesn’t always happen on the first try

• One previous Programming Assignment or Creative Project can be resubmitted each week
  • Must be accompanied by a write-up describing changes (via Google Form)
  • Grade on resubmission will replace original grade
  • Each assignment should only be resubmitted once
  • You may only submit assignments <4 weeks old

• We will drop your two lowest quiz problem grades
  • No special action required– we’ll do this automatically

• See the syllabus for more details
Grading

Grades should reflect your proficiency in the course objectives

• All assignments will be graded E (Excellent), S (Satisfactory), or N (Not yet)
  • Under certain circumstances, a grade of U (Unassessable) may be assigned
  • In some cases, not all grades will be given
• Final grades will be assigned based on the amount of work at each level

• See the syllabus for more details
Agenda

• About us
• About this course
  • Learning objectives
  • Other similar courses
  • Course components
• Our learning model

• Tools and resources
  • Course Website
  • Ed
  • VS Code
• Assessment and grading
  • Collaboration
Collaboration Policy

*Learning is hard, but it’s easier when you learn from each other*

• You are encouraged to form study groups; work together on pre-class work, practice and review; and discuss your ideas and approaches
• All work you submit for grading **must be predominantly and substantially your own**
• Work that violates policy may be withdrawn within 72 hours

• See the [syllabus](#) for more details