Welcome to CSE 121!

Use this QR code as one way to ask questions!

Matt Wang
Spring 2024

TAs: Andy Anju Archit Arkita Autumn Christian
Hannah H Hannah S Heather Hibbah Janvi Jessie
Jonus Julia Luke Maria Mia Ritesh
Shayna Simon Trey Vidhi Vivian Gumball?

sli.do #cse121-0

Today’s playlist: CSE 121 lecture beats 24sp
Agenda (1/7)

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

• Tools and resources
  • Course Website
  • Ed
• Assessment and grading
• Collaboration
Agenda (2/7)

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

• Tools and resources
  • Course Website
  • Ed
• Assessment and grading
• Collaboration
Hi, I’m Matt! (he/him)

- (new) Assistant Teaching Professor in the Allen School
- grew up mostly in Toronto and sometimes Tokyo!
- went to UCLA!
  - BS & MS in Computer Science
  - BS in Math-Economics
- computer science interests: CS education, “open-source”, programming languages, accessibility
- non-CS interests: reading, music (Laufey was my #1 this wrapped), video games, skiing & ice skating!
Meet your 23 TAs!
Agenda (3/7)

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

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

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

• Computational Thinking
• Code Comprehension
• Code Writing
• Communication
• Testing
• Debugging
• Ethics & Societal Impact
## Other Similar Courses

<table>
<thead>
<tr>
<th>Course</th>
<th>Good choice if...</th>
</tr>
</thead>
</table>
| CSE 121 | • You’ve never programmed before AND  
• You are, or want to be, in a major such as CS, CE, ECE, Info, etc. that requires Java programming                                   |
| 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 123 | • You’ve taken CSE 122 AND  
• You are, or want to be, in a major such as CS, CE, ECE, Info, etc. that requires Java programming                                   |
| CSE 160 | • You’ve never programmed before AND  
• You’re interested in data science and analysis OR  
• You’d rather learn Python than Java* OR  
• You are, or want to be, in a major such as Physics, Bio, Stat, etc. where analyzing data through programming is useful |
Course Components

Meetings

- LECTURES (x20)
  - We’re here!
  - Introduce concepts, practice ideas, discuss applications.
  - Pre-class materials to prepare for class each day. Due before class.

- SECTIONS (x16)
  - Held in person
  - More practice, review, applications
  - TA advice, how to be an effective student
  - Preparation for quizzes / exams

Assessments

- PROGRAMMING ASSIGNMENTS (x4)
  - Structured assignments
  - Programming in Java
  - Applying & implementing course concepts

- CREATIVE PROJECTS (x4)
  - More open-ended assignments
  - Explore new ideas and applications

- QUIZZES (x3)
  - Taken in quiz section
  - 45 minutes on computer

- EXAM (x1)
  - Culminating exam
  - Wed, June 5th
    2:30 – 4:20 PM

Lesson 0 - Spring 2024
Agenda (4/7)

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

• Tools and resources
  • Course Website
  • Ed
• Assessment and grading
• Collaboration
How Learning Works

Learning requires **active participation** in the process.

It’s not as simple as sitting and listening to someone talk at you!

- Requires **deliberate practice** in **learning by doing**
- Benefits from **collaborative learning**
- **Does not work well if you cram everything!**
Pre-Class Materials (1/3)

Core element of course: **pre-class material**

- prepare for each lecture with readings & practice problems
- should take ~30 minutes per lecture (why we don’t have Monday lectures!)
- class will start with a brief recap, then pick off where we left off
Pre-Class Materials (2/3)

Core element of course: **pre-class material**

- prepare for each lecture with readings & practice problems
- should take ~30 minutes per lecture (why we don’t have Monday lectures!)
- class will start with a brief recap, then pick off where we left off

Which means...

- we can spend lecture diving deeper, answering questions, and think-pair-share
- you can ask about pre-lecture material in class or quiz section!

Lesson 0 - Spring 2024
Pre-Class Materials (3/3)

Core element of course: **pre-class material**
- prepare for each lecture with readings & practice problems
- should take ~30 minutes per lecture (why we don’t have Monday lectures!)
- class will start with a brief recap, then pick off where we left off

Which means…
- we can spend lecture diving deeper, answering questions, and think-pair-share
- you can ask about pre-lecture material in class or quiz section!

Pre-class materials are ungraded, which means…
- it’s okay if you find them challenging – that means you’re learning!
- but, **you should do them**, and **we will assume you’ve done them**
Consistent and Active Participation (1/2)

Attendance is not graded. But, it’s strongly encouraged!

- lectures & sections are not going to be just us talking at you!
- ex: live in-class coding, debugging, think-pair-share, and problem-solving
- spreading out ~ 1-2 hours each day over Tuesday – Friday is much more effective than cramming before the assignment is due!
Consistent and Active Participation (2/2)

Attendance is not graded. But, it’s strongly encouraged!

- lectures & sections are not going to be just us talking at you!
- ex: live in-class coding, debugging, think-pair-share, and problem-solving
- spreading out ~ 1-2 hours each day over Tuesday – Friday is much more effective than cramming before the assignment is due!

Catching up:

- all lectures are recorded on Panopto; slides are on our website.
- section materials are on Ed, but section will not be recorded.
Metacognition

- **Metacognition**: asking questions about your solution process.

Examples:
- **While debugging**: explain to yourself why you’re trying this change.
- **Before running your program**: make an explicit prediction of what you expect.
- **When working**: be aware when you’re not making progress, so you can take a break or try a different strategy.
- **When designing**:
  - Explain the tradeoffs with using a different data structure or algorithm.
  - If one or more requirements change, how would the solution change as a result?
  - Reflect on how you ruled out alternative ideas along the way to a solution.
- **When studying**: what is the relationship of this topic to other ideas in the course?
Learning in CSE 121 (or anywhere)
Course Culture and Support

• Currently 231 students enrolled!
  • *Almost none* are CSE majors!
  • Wide range of backgrounds, interests, and goals
  • *Everyone* is new to programming
• Support and help each other!
  • Form study groups
  • If you have a question, others almost certainly do too
Course Culture and Support: Live Support

Introductory Programming Lab (TA Office Hours – starting Week 2)
- #1 place to get help (and highly rated in the class!)
- face-to-face help from TAs on any course questions – not just assignments

TA Section
- Work through practice problems (this is how you learn!)
- Get to know your TAs & peers!

Instructor Office Hours (in-person & Zoom – schedule on website)
- I don’t byte (most of the time)
- Great for things from lecture, personal questions, or just to say hi!
Course Culture and Support: Ed & Email

Ed Board
- Best for content and logistics questions – 231 of you >> 24 of us!!
- Feel free to make them public or private (and/or anonymous)
- Answer other students’ questions – great way to learn!

Email
- Best for personal circumstances and/or private questions
- If unsure, always feel free to email Matt (at mxw@cs.washington.edu)
- May politely ask you to post on Ed instead!
The World Around CSE 121 & Reaching Out

Our goal is to give you a great CSE 121 experience!

But CSE 121 does not exist in a vacuum – there’s a lot going on in the world right now that can impact your education.

We’ve designed course policies for maximum flexibility: resubmissions, dropping quiz/exam problems, asynchronous help & lecture recordings.

Please reach out ASAP if you’re struggling or have circumstances that require extra support. We’re happy to help – we just need to know!
Agenda (5/7)

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

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

• Primary source of course information (not Canvas)
• Calendar will contain links to (almost) all resources
Syllabus (website)

Please review the syllabus ASAP.
Ed

- Our online learning platform
- Lessons, sections, quizzes all here
- Intro and walkthrough in Section 0
Other Course Tools (brief overview)

My Digital Hand
- Queueing in office hours

Visual Studio Code
- Not strictly necessary!
- Develop offline
- Debugger Tool

Canvas / Panopto
- Lecture recordings

Sli.do
- In-class activities (ungraded)
- No account needed
Agenda (6/7)

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

• Tools and resources
  • Course Website
  • Ed
• Assessment and grading
• Collaboration
Assessment and Grading

• Our goal in the course is for you to gain proficiency of 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
Grading

*Grades should reflect proficiency in course objectives.*

All assignments will be graded with “E/S/N” grading:
- E (Excellent)
- S (Satisfactory)
- N (Not Yet)

Final grades are assigned based on amount of work at each level.

We’ll discuss this more when our first assignment is released. See [syllabus](#) for more details.
Learning takes time, and doesn’t always happen on the first try!

Each week, one previous assignment or project can be resubmitted.
- Must be accompanied by write-up explaining change (reflection!)
- Grade on your resubmission replaces original grade
- Assignments are only eligible for resubmission within 3 “cycles” following its grade being released

We’ll discuss this more when our first assignment is graded. See syllabus for more details.
Agenda (7/7)

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

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

You are **encouraged** to form study groups, work together on practice and review, and discuss your ideas & approaches at a high level.

- Sharing ideas and working together is good – but, please **cite them**
- Don’t copy work. In particular, **never send someone else your code**.

All work you submit must be **predominantly** and **substantially** your own.
- Includes Generative AI tools! (see [dedicated website page](#))

See [syllabus](#) for more details, including on the withdrawal policy.
Help us improve!

CSE 121 is super new! We’ve worked hard to build a course that we think will be effective, supportive, and help you succeed.

But... 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
- Mid and end-of-quarter feedback
- Use CSE Anonymous Feedback Tool
“Homework” for Next Time

First assignment will be released Friday, but there are some things to do in the meantime.

TODOs this week:

- Fill out the introductory survey
- Go meet your TA and classmates in Thursday’s quiz section
- Complete the pre-class material for Friday (see calendar)
- Check over syllabus details