

LEC 00

CSE 121

Welcome!



Questions during Class?

Raise hand or send here

sli.do #cse121



BEFORE WE START

*Talk to your neighbors:
Introduce yourself!*

*What is your name? Major?
What would you be doing if you
didn't have class right now?*

Instructor: Miya Natsuhara and Anjali Pal

TAs:	Samrutha	Tamsyn	Owen	Jennie
	William	Cedric	Jameson	Anisha
	Samvedha	Anum	Paul	TJ
	Syreeta	Navya	Gurbax	Ivory
	Ryan	Gene	Tony	Sarah W
	Anant	Margot	Lucas	Avery
	Henry	Sarah K	Alex	
	Reese	Ethan	Jesse	
	Savannah	Evelyn	Matthew	

Lecture Outline

Today:

1. Introductions :)
2. About this course
3. Our learning model
4. Culture and community
5. Tools
6. Our first program!

On Friday:

1. Assessment and grading
2. Collaboration
3. More programming!

Hi, we're Miya (she/her) and Anjali (she/her)!

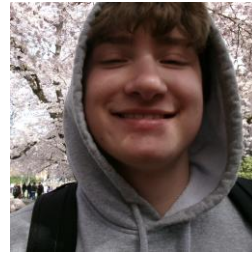
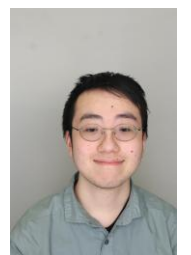
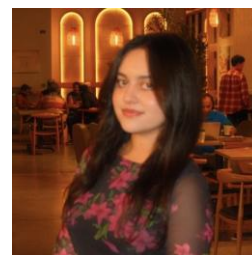
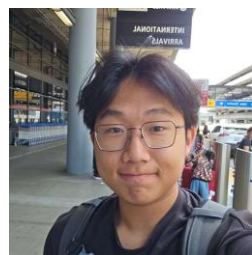
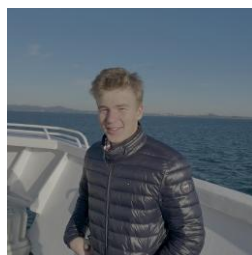
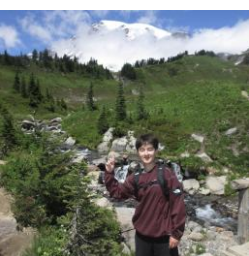
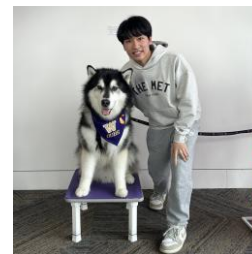
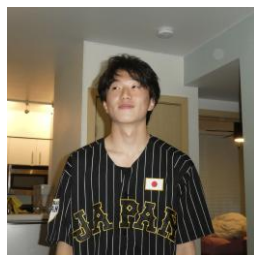
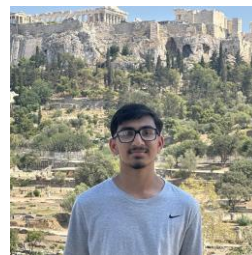
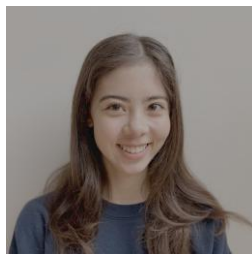
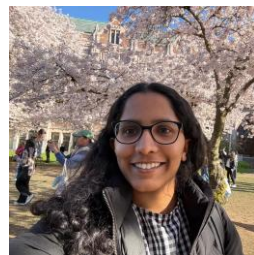


- Assistant Teaching Professor
- Frequent intro CS instructor
- Works in areas of CS Education, Accessibility
- Non-CS hobbies: knitting, baking, playing with my corgi



- 5th year Ph. D. student
- First time teaching intro!
- Works in Programming Languages, Verification
- Non-CS hobbies: hiking, frisbee, pottery

Meet your TAs!



Agenda: About This Course

Today:

1. Introductions :)
2. **About this course**
 - Learning objectives
 - Similar courses
 - Course components
3. Our learning model
4. Culture and community
5. Tools
6. Our first program!

On Friday:

1. Assessment and grading
2. Collaboration
3. More programming!

Learning Objectives

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

Bottom line:

Intro to Programming, part 1

Not quite:

- “How do computers work?”
- “Intro to the Java language”
- “All you need to program”
- Math!

Learning Objectives:

1. Computational Thinking
2. Code Comprehension
3. Code Writing
4. Communication
5. Testing
6. Debugging
7. Ethics & Societal Impact

Other Similar Courses

Course	Good choice if...
CSE 121 (this is us!)	- 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 <i>some</i> 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 (or equivalent) 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, through UWHS, or at community college) OR - You took AP CS A (or similar) and feel confident in <i>all</i> the material
CSE 143X	- You have programmed quite a bit before, but <i>not</i> in Java OR - You have lots of extra time to put into learning and tend to pick things up quickly
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 your primary goal is analyzing data through programming (rather than building software)

Also see: [guided self-placement](#) and [CSE page on introductory courses](#) for more info.

Course Components

LECTURES

x16

- We're here!
- Introduce concepts, practice ideas, discuss applications.
- Pre-class materials to prepare for class each day. Due **before** class.

SECTIONS

x14

- Held in person
- More practice, reviews, applications
- TA advice, how to be an effective student
- Preparation for quizzes

Meetings

PRE-CLASS MATERIALS

x~15

- Short lessons introducing concepts to be discussed in lecture that day
- Due **before** each lecture

COMPUTER QUIZZES

x4

- 45 minutes, scheduled individually
- Builds on the programming projects

PROGRAMMING PROJECTS

x4

- Structured assignments
- Programming in Java
- Applying & implementing course concepts

PAPER QUIZZES

x4

- Taken in quiz section/final exam period
- 45 minutes on paper

Graded Assignments

Agenda: Our Learning Model

Today:

1. Introductions :)
2. About this course
3. **Our learning model**
 - Learning objectives
 - Similar courses
 - Course components
4. Culture and community
5. Tools
6. Our first program!

On Friday:

1. Assessment and grading
2. Collaboration
3. More programming!



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**
- Involves **productive struggle**
- Benefits from **collaborative learning**
- Does not work well if you cram everything!



Learning and AI: Our Philosophy

AI amplifies the skills of the AI user

- We cannot teach an introductory programming course without acknowledging the **existence of AI tools**.
- We strongly believe in the value of learning the fundamentals of programming.
- The ability to **understand** and **modify** code is more important than ever.
- AI use is **permitted** on some assignments. AI use is **not required**.

Pre-Class Materials

PRE-CLASS MATERIALS

PCMs are a core element of the course

- 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

This means...

- It's okay if you find them challenging – that means you're learning!
- We can spend lecture diving deeper, answering questions, and think-pair-share
- You can ask about pre-lecture material in class or section!

Consistent and Active Participation

ATTENDANCE

- Lectures and sections are not going to be just us talking at you!
- Instead: live in-class coding, debugging, think-pair-share, and problem—solving
- Spending ~1-2 hours each day over Tuesday - Friday is **much more effective** than cramming right before a quiz!

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** (~troubleshooting): reflect on *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**:
 - What are the hard parts of the problem going to be?
 - 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?

We'll ask you to practice metacognition regularly throughout the course

Learning in CSE 121: Live Support Systems

Programming is hard! We **want** to give you collaborative support!

Introductory Programming Lab (TA Office Hours) – starting Week 2

- *many* hours/week (and highly rated in the class!)
- Face-to-face help from TAs on **any** course questions

Instructor Office Hours – starting this week!

- Great for things from lecture, personal questions, or just saying hi

Learning in CSE 121: Async Support Systems

Ed Board

- Best for content and logistics questions – 500+ of you!!
- Encourage public posts, except for things about **your** graded work
- Answer other students' questions – great way to learn!

Email

- Best for personal circumstances and/or private questions
- Unsure? Feel free to email Miya and Anjali (both of us please!)
 - cse121-instructors@cs.washington.edu
- May politely ask you to post on Ed instead!
- For emails, **please use your UW email** (protecting student privacy!)

Agenda: Culture and Community

Today:

1. Introductions :)
2. About this course
3. Our learning model
4. **Culture and community**
5. Tools
6. Our first program!

On Friday:

1. Assessment and grading
2. Collaboration
3. More programming!



Practice: Think

[sli.do](#)[#cse121](#)

Think-Pair-Share: Inclusive Environments

CSE 121 will have many think-pair-share activities. Let's practice! Today's think:

What was an experience you had that made you feel welcome or included in a learning environment?

1. **Think** on your own, in silence for about ~ 30 seconds
2. **Pair** with your neighbor about it (and introduce yourself!!)
3. **Share** in sli.do & in class (I'll take a few volunteers from both)



Practice: Think

sli.do

#cse121

Think-Pair-Share: *Exclusive* Environments

CSE 121 will have many think-pair-share activities. Let's practice! Today's think:

What was an experience you had that made you feel unwelcome or excluded in a learning environment?

1. **Think** on your own, in silence for about ~ 30 seconds
2. **Pair** with your neighbor about it (and introduce yourself!!)
3. **Share** in sli.do & in class (I'll take a few volunteers from both)

The CSE 121 Community

ABOUT US

- Currently ~580 students enrolled!
 - *Very few* are CSE majors
 - Wide range of backgrounds, interests, and goals
 - ***Everyone*** is new to programming

OUR GOALS

- Foster an **inclusive** and **supportive** environment for all students to thrive by:
 - being respectful
 - being kind and understanding
 - being honest
 - being ourselves

The World Around Us

College is challenging and CSE 121 isn't your only class.

Life is unpredictable and things happen.

We can't leave the impacts of the world around us at the classroom door.

OUR POLICIES

Our course policies are **designed for flexibility**:

- Asynchronous help
- Lecture recordings

SUPPORT

We're here to **support you as a student and as a person**.

Please **reach out** if you're struggling or have circumstances that require extra support.

Help Us Improve!

This is still a relatively new course! We're *always* looking for feedback on how to improve the class for you and for future students.

- We ***really*** value your feedback!
- Let us know what's working and what isn't working for you!

Several feedback mechanisms:

- Built into the class (e.g. reflections, mid-quarter feedback sessions)
- Post on discussion board (can be public/private)
 - Note: anonymous is anonymous to other students, *not* to staff
- Use [CSE's Anonymous Feedback Tool](#) (also on website)

Agenda: Tools

Today:

1. Introductions :)
2. About this course
3. Our learning model
4. Culture and community
5. **Tools**
 - Course website
 - Ed
 - Other Key Course Tools
6. Our first program!

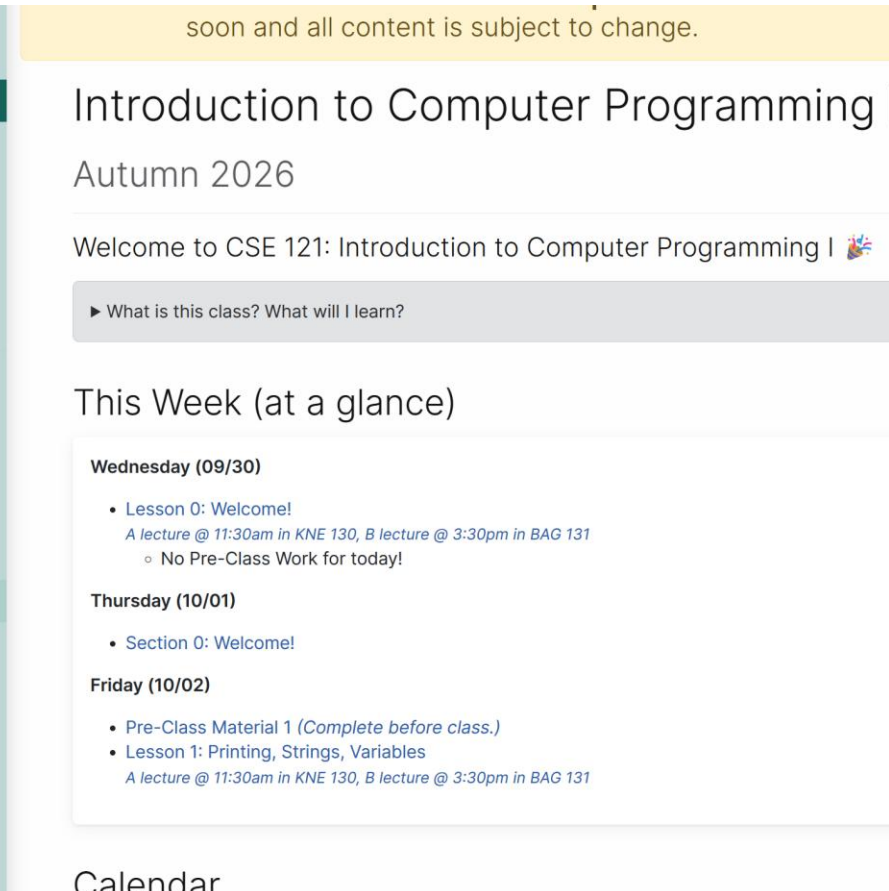
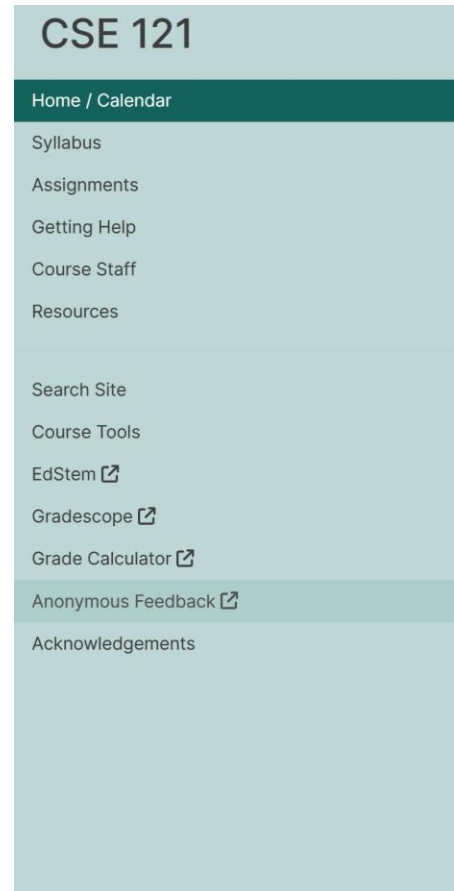
On Friday:

1. Assessment and grading
2. Collaboration
3. More programming!

Course Website

cs.uw.edu/121

- Primary source of course information (**not Canvas**)
- Calendar will contain links to (almost) all resources
- Please **review syllabus ASAP**
- Let's go on a website tour :)



Ed

- Our online learning platform
- Lessons, sections, announcements
- Place to ask questions
- Also, **where we'll code!**
- Intro and walkthrough in **Section 0**

The screenshot shows the Ed platform interface for CSE 121 - 26au. The top navigation bar is purple with the University of Washington logo and the course title. Below the navigation bar, there's a search bar and a 'New Thread' button. A list of threads is displayed, with the first thread titled 'Welcome to CSE 121!' by Miya Natsuhara, an instructor/lead TA. The thread has 10 likes and 2 comments. The thread content includes a welcome message, a link to 'EdStem's Quick Start Guide', and a section titled 'Updates for Autumn 2026' which mentions significant changes to the course. At the bottom, there's a 'Resources' section with a book icon.

ed W UNIVERSITY of WASHINGTON CSE 121 - 26au – Ed Discussion

New Thread

Search

Filter

Welcome to CSE 121! #1

Miya Natsuhara INSTRUCTOR/LEAD TA 8 hours ago in Announcements

Hi all!

Welcome to CSE 121! 🍁! My name is [Miya Natsuhara](#), and I'll be co-instructor for CSE 121 in Autumn 2026 alongside my co-instructor [Anjali Pal](#). The entire course will be with you in CSE 121! With the quarter officially starting tomorrow, I want to get some information out so you can prepare.

We hope you all had an enjoyable summer. You're receiving this announcement because Ed will be one of the main places for you to connect with your classmates and instructors. If you want to Ed or want a refresher, check out [EdStem's Quick Start Guide](#) to get familiar with the platform.

This is a long message with a lot of information. Some of it will be new to you, so please try to at least skim this whole announcement, and pay particular attention to the updates for Autumn 2026.

🌟 Updates for Autumn 2026

We are making a number of significant changes to the way CSE 121 is taught this whole, operates for this year. The CSE 12X teaching team has written up these changes and the philosophy behind them. We will discuss the specifics in class, but please read the [vision statement](#) before class to have a sense of what to expect.

📖 Resources

Other Key Course Tools



sli.do

- Ask questions in class
- Live activities (ungraded)
- No account needed



Canvas

- Panopto lecture recordings (also linked from website)



Gradescope

- Quiz and final exam grading
- Pre-section work

more course tools introduced soon™

Switching to Ed: Our First Program!*

*note: in almost all cases, slides are *not* comprehensive. reviewing the slides will not cover all the content in lecture!

“Homework” for Next Time

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

TODOs this week:

- Fill out the [introductory survey](#) (this is part of your pre-section work for tomorrow's section)
- Go meet your TA and classmates in Thursday's quiz section
- Complete the pre-class material for Friday (see website/calendar)
- Check over [syllabus details on website](#)