

LEC 00

CSE 123

Welcome & Syllabus



Questions during Class?
Raise hand or send here

sli.do #cse123



BEFORE WE START

Talk to your neighbors:
Introduce yourself to your neighbor!

*What is your name? Major? What
have you been up to the past week?*


Instructors: Brett Wortzman
Miya Natsuhara

TAs:

Arohan	Neha	Rushil	Johnathan	Nicholas
Sean	Hayden	Srihari	Benoit	Isayah
Audrey	Chris	Andras	Jessica	Kavya
Cynthia	Shreya	Kieran	Rohan	Eeshani
Amy	Packard	Cora	Dixon	Nichole
Trien	Lawrence	Liza	Helena	

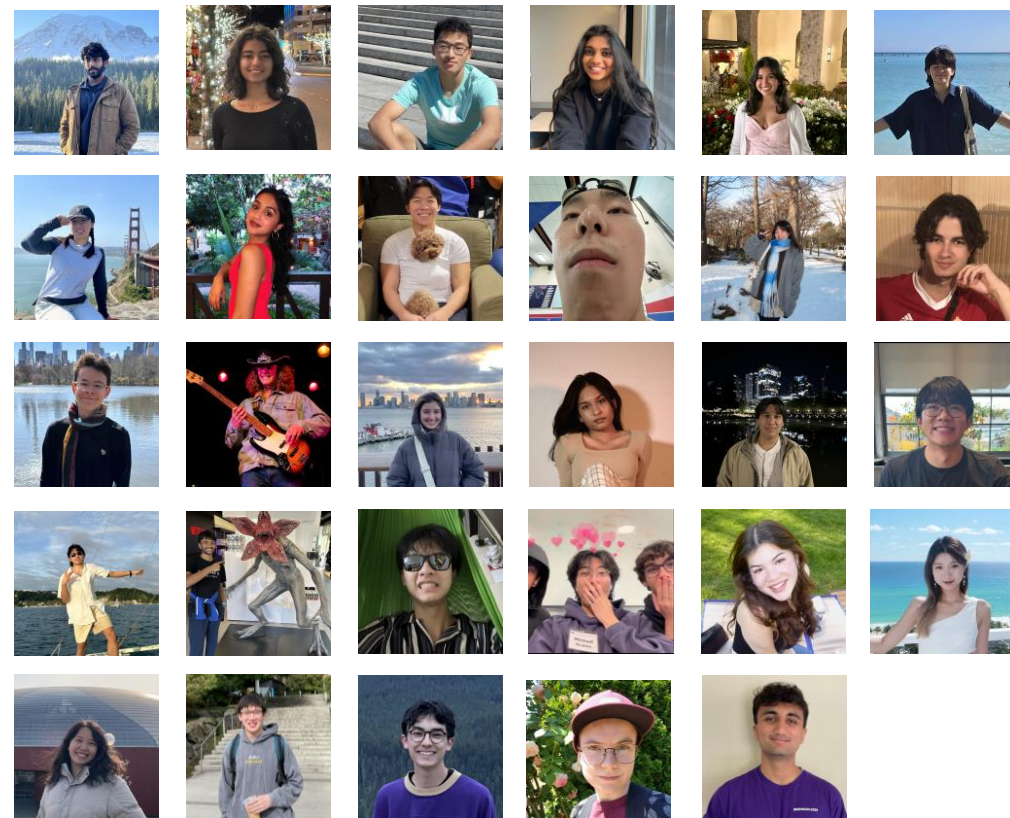
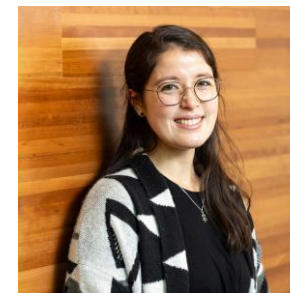
Music: [CSE 123 25wi Lecture Tunes](#)

Lecture Outline

- **Introductions** 
- About this Course
 - Course Components & Tools
 - Making the Most of this Class
- Assignments and Grading

Course Staff

- Instructors: Brett Wortzman
Miya Natsuhara
- Teaching Assistants: [29 Fantastic TAs!](#)
 - Available in section, office hours, and discussion board
 - Invaluable source of information & help in this course
- We're excited to get to know you!
 - Our goal is to help you succeed 😊



What is this Class?

CSE 121 – Computer Programming I

- Data types (int, String, boolean)
- Methods / Functions
 - Parameters, Returns
- Control structures
 - Loops, Conditionals
- Arrays & 2D arrays
- **Computational Thinking**
(language agnostic)

CSE 122 – Computer Programming II

- Functional Decomposition
- File I/O
- Using data structures
 - List, Stacks / Queues, Sets, Maps
- Object Oriented Programming
 - Interfaces

CSE 123 – Computer Programming III

- Advanced Object Oriented Programming
 - Comparable, Inheritance/Polymorphism, Abstract Classes
- Implementing data structures
 - ArrayLists, LinkedLists, Trees
- Recursion
- Critical analysis of design

Why 123?

1. To solve more complex problems by leveraging more complex programming structures / patterns

2. To better rationalize specific design decisions

- How to “best” structure programs
- Which data structures are “most” appropriate to use

3. To understand and critically analyze intersections between Computer Science and society

- Search engines, algorithmic art, machine learning, etc.
- Developing informed opinions on current issues

Be a better programmer

Be a better person

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

What do you want to get out of this course?

sli.do #cse123



Lecture Outline

- Introductions
- **About this Course**
 - **Course Components & Tools** 
 - Making the Most of this Class
- Assignments and Grading

Course Website

cs.uw.edu/123

Syllabus

Course Information

Teaching Staff
Instructor: Brett Wortzman & Miya Natsuhara
Instructor Email: cse123-instructors@cs.washington.edu
Registration Questions: CSE Advisers (ugrad-adviser@cs.washington.edu)
Course Staff and Support Hours: [Course Staff and Office Hours](#)

▼ Who to contact?

Here are some common types of questions and the best place to ask them to get the fastest and most accurate response.

- **Registration questions?** Email the [CSE advisers](#) as the course staff do not have access to add codes.
- **Questions about course concepts?** Visit [office hours](#) in the introductory...

1) Course Information

2) Course Goals

2.1) Learning Objectives

3) Software and Textbooks

4) Class Sessions and Quiz Sections

4.1) Class Sessions

4.2) Quiz Sections

5) Inclusion

6) Required Course Work

6.1) Types of Assignments

6.2) Late Work

7) Course Support Structures

Review the syllabus!

CSE 123

- Home / Calendar
- Syllabus
- Assignments
- Exam
- Staff
- Office Hours
- COVID-19 Safety
- Resources
- Course Tools [↗](#)
- EdStem
- Anonymous Feedback
- Grade Checker

Attention! This website is still **under development**. More information will be added soon and all content is subject to change.

Introduction to Computer Programming

Winter 2025

Welcome to CSE 123: Introduction to Computer Programming III 🇺🇸

- ▶ What is this class? What will I learn?
- ▶ Prior Experience and Expectations

Syllabus If you want to learn more about the course and its policies, please check out our [course syllabus](#).

Feedback Feedback is always welcome! You can contact the [course staff](#) or submit anonymous feedback.

Registration Please **do not** email the course staff or instructors regarding registration for the course. The course staff do not have access to add codes. Please email ugrad-adviser@cs.washington.edu for assistance.

Announcements

This Week (at a glance)

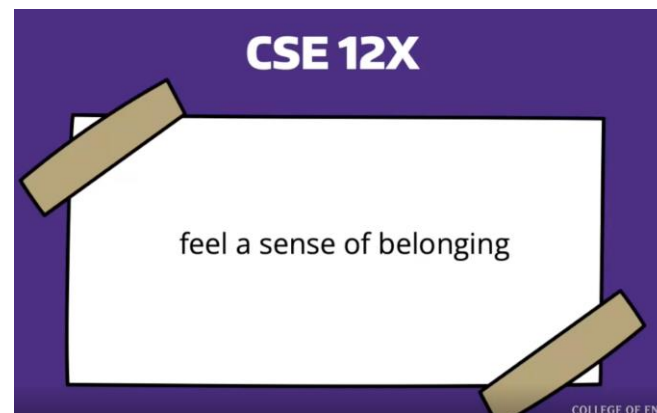
Monday (01/06)

- Nothing!

Contains most course info – check frequently!
 Announcements, Calendar, Lecture Slides, Office Hours schedule,
 Staff Bios, Important Links

Creating an inclusive environment

Video



- This is a more professional environment than hanging out with friends
- Think about the impact your words can have.
- Collaboration, Support, and Empathy
- Check your own biases and communicate thoughtfully
- Challenge unacceptable behaviors

Other Course Tools



Ed

- Community & Information
 - Discussion Board
(please ask & answer!; anonymous option)
 - Announcements
- Pre-Class Materials / Section Handouts
- Assignments
 - Online IDE
 - Submit assignments
 - View Feedback

My Digital Hand



My Digital Hand

- Queueing in office hours

VSCode

- Develop offline
- Visual debugger



Canvas


- Lecture recordings



Sli.do

- In-class activities
(ungraded)
- No account needed

Lecture Outline

- Introductions
- **About this Course**
 - Course Components & Tools
 - **Making the Most of this Class** 
- Assignments and Grading

Digression: Brett's Pandemic Hobby

Amigurumi: Japanese art of creating crocheted or knitted stuffed toys



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**
- Hybrid classroom model
 - Asks you to do some preparation before class in the form of readings and practice problems.
 - Should take ~30 minutes outside of class per lesson
 - Class will start with brief recap, then pick up where the reading and practice problems leave off.
 - Attendance isn't graded, but showing up and trying is the first step in succeeding in the class!
- Pre-class materials are ungraded, but...
 - It's okay if you find them challenging! That means you are learning!



Getting Help

- Discussion Board
 - Feel free to make a public or private post on Ed
 - We encourage you to answer other peoples' questions! A great way to learn
- Introductory Programming Lab (Office Hours)
 - TAs can help you face to face in office hours, and look at your code
 - You can go to the IPL with **any** course questions, not just assignments
- Section
 - Work through related problems, get to know your TA who is here to support you
- Your Peers
 - We encourage you to form study groups! Discord or Ed are great places to do that
- Email
 - We prefer that all content and logistic questions go on the Ed discussion board (even if you make them private). Many more students than staff!
 - For serious personal circumstances, you can email Miya/Brett directly. It never hurts to email us, but if it's a common logistic question, we may politely ask you to post on the discussion board instead.

Lecture Outline

- Introductions
- About this Course
 - Course Components & Tools
 - Making the Most of this Class
- **Assignments and Grading** ◀

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 (Tuesday, March 18)
 - Final, culminating assessment of all your skills and knowledge

Resubmission and Ignored Quiz Problems

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
 - An assignment can be resubmitted in the 3 cycles after feedback has been published
 - *Tip: Resubmit as early as possible!*
- We will ignore 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
- Final grades will be assigned based on the **amount of work at each level**
- See the [syllabus](#) for more details

Collaboration Policy





- When we assess your work in this class, we need to know that it's *yours*.
- Unless otherwise specified, **all graded work must be completed individually**.

Some specific rules to highlight:

- do not share your own solution code or view solution code from any source – including but not limited to other students, tutors, or the internet
- do not use AI tools (e.g. ChatGPT) on graded work in any capacity

See the [syllabus](#) for more details (this is *very* important to understand).

Coming up...

-  Go to your first quiz section and meet your TAs tomorrow!
 - Make sure to double check [MyUW](#) for the location.
-  Complete Pre-Class Work 1 before class on Friday!
 - It will be posted and linked from the course calendar later today.
-  Complete the [Introductory Survey](#)
 - This helps us gather data about the students taking our classes and their backgrounds, to inform future offerings.
-  Consider attending the Review Session on Monday, Jan 13
 - 12:30pm – 1:20pm in ARC 147 ; 2:30pm – 3:20pm in GUG 220
 - Optional, hopefully recorded (waiting for confirmation)