

LEC 00

CSE 121

Welcome!



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: Hayden Feeney

TAs: Anisha Cayden Savannah Tamsyn William

Questions during Class?

Raise hand or send here

sli.do **#cse121**



Lecture Outline

Today:

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

On Friday:

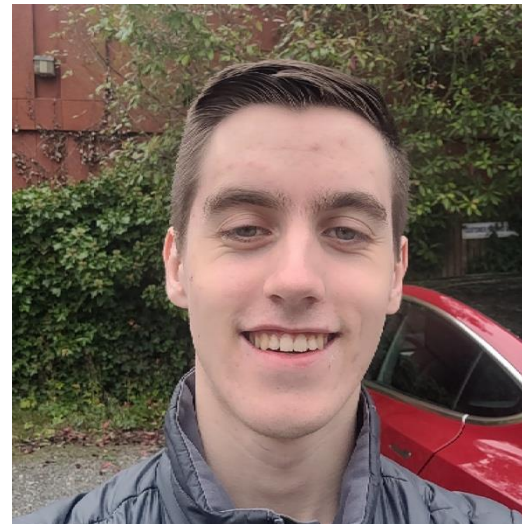
1. More programming!

Hi, I'm Hayden! (he/him)

- First time instructor
- 10 quarters as a TA (121, 123)
- Recent MS graduate from UW CSE
- Hometown Tacoma WA!
- Some interests:
 - Dancing (salsa, lindy hop, others!)
 - Education, community, interpersonal relationships + mental health
 - Cooking
 - Languages



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. Assessment and grading
7. Collaboration
8. Our first program!

On Friday:

1. 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 / exams
- Pre-section work due at the beginning of section each day

Meetings

PROGRAMMING ASSIGNMENTS

x4

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

QUIZZES

x3

- Taken in quiz section
- 55 minutes on paper

CREATIVE PROJECTS

x4

- More open-ended assignments
- Explore new ideas and applications

EXAM

x1

- Culminating exam
- **Fri, Aug 21st**
- 12:00 – 1:00 PM

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. More programming!
2. Assessment and grading
3. 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**
- Involves **productive struggle**
- Benefits from **collaborative learning**
- Does not work well if you cram everything!



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...

- We can spend lecture diving deeper, answering questions, and think-pair-share
- You can ask about pre-lecture material in class or 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

ATTENDANCE

Attendance is not graded, but it's strongly encouraged!

- 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 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** (~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 – 35 of you > 6 of us!!
- 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 Hayden (hbfeeney@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. Assessment and grading
7. Collaboration
8. Our first program!

On Friday:

1. 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 ~35 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**:

- Resubmissions
- Dropping quiz/exam problems
- 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. Assessment and grading
7. Collaboration
8. Our first program!

On Friday:

1. 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 :)

The screenshot displays the CSE 121 course website. On the left is a sidebar menu with the following links: Home / Calendar (highlighted), Syllabus, Assignments, Resubmissions, Exam, Getting Help, Course Staff, Grading Rubrics, Resources, Search Site, Course Tools, EdStem, Gradescope, Grade Calculator, Anonymous Feedback, and Acknowledgements. The main content area on the right is titled 'CSE 121 Introduction to Computer Programming I Summer 2026'. Below the title is a yellow banner with the text: '26su ver. Note: this is for the Summer 2026 iteration of CSE 121. Looking for a different quarter? Please visit <https://courses.cs.washington.edu/courses/cse121/>'. Below the banner is a welcome message: 'Welcome to CSE 121: Introduction to Computer Programming I' with a small icon. A grey button with the text '► What is this class? What will I learn?' is located below the welcome message. At the bottom of the main content area is a section titled 'This Week (at a glance)' which lists the schedule for the week of June 22nd to 24th. The schedule shows that Monday (06/22) and Tuesday (06/23) have 'Nothing!' planned, while Wednesday (06/24) has 'Lesson 0: Welcome, Printing' at 12:00pm in SIG 134.

CSE 121

Home / Calendar

Syllabus

Assignments

Resubmissions

Exam

Getting Help

Course Staff

Grading Rubrics

Resources

Search Site

Course Tools

EdStem

Gradescope

Grade Calculator

Anonymous Feedback

Acknowledgements

Introduction to Computer Programming I

Summer 2026

26su ver. Note: this is for the Summer 2026 iteration of CSE 121. Looking for a different quarter? Please visit <https://courses.cs.washington.edu/courses/cse121/>

Welcome to CSE 121: Introduction to Computer Programming I

► What is this class? What will I learn?

This Week (at a glance)

Monday (06/22)

- Nothing!

Tuesday (06/23)

- Nothing!

Wednesday (06/24)

- Lesson 0: Welcome, Printing
A lecture @ 12:00pm in SIG 134

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 displays the Ed platform interface for CSE 121. The top navigation bar includes the 'ed' logo, the University of Washington name, the course title 'CSE 121 - ...', and a 'New Thread' button. Below the navigation bar, there is a search bar and a list of threads. The first thread, titled 'Welcome!', is highlighted and shows it was posted 23 hours ago by Hayden Feeney (Instructor / TA) with 2 new replies and 3 likes. The main content area shows the 'Welcome! #1' post by Hayden Feeney. The post includes a heart icon, a reply count of 3, and the text: 'Hi everyone! My name is Hayden Feeney, and I'll be your instructor for CSE 121 this quarter. The rest of the course staff and I are looking forward to working with you! We wanted to welcome you to the course and provide you with information so you're feeling prepared for this first week of classes. This announcement is being sent through EdStem, which is our course's platform for discussion, announcements, and more. If you're new to EdStem or would like a refresher on how to use it, please feel free to check out [EdStem's Quick Start Guide](#). The rest of this post is a combination of different logistics items. **Please read this carefully before our first class on Wednesday (two days from now)!**' Below the post, there is a 'Resources' section with a book icon and the text: 'Our [course website](#) has all the information and resources you'll need and will be'.

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™

Agenda: Culture and Community

Today:

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

On Friday:

1. More programming!

Assessment

Why assessment?

- Our goal: you gain **proficiency of the concepts and skills** we teach
- We *assess* your proficiency by asking you to apply these concepts and skills on tasks
- By necessity, we are assessing your work as a proxy for proficiency

Resubmissions

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 reflection explaining your change(s)
- Grade on resubmission replaces original grade
- Assignments eligible for only 3 “cycles” after feedback released

We'll discuss more after our first assignment is graded. See [syllabus](#)!

Section Credit

A “nudge” system: earn **section credit** by attending and engaging in your quiz section.

If you **earn section credit for 9 of the 13 sections**, you'll get an extra resubmission!

- no direct grade impact, think of it as optional/extra credit

Grading Scheme

Grades should reflect proficiency in course objectives.

All assignments, quizzes, and exams are graded with an “E/S/N” grade:

- **E (Excellent)**
- **S (Satisfactory)**
- **N (Not Yet)**
- **[U (Unassessable)]**

Grading Assessments

E/S/N (or ESN) grades per assessment type:

- **Programming Assignments**: 4 ESN grades (4 assignments, 16 total)
- **Creative Projects**: 1 ESN grade (4 projects, 4 total)
- **Quizzes**: 3 ESN grades (3 quizzes, 9 total)
- **Final Exam**: 6 ESN grades

We will also drop (ignore) your lowest 2 quiz/final exam grades.

Course Grades

We provide a “***minimum grade guarantee***” for translating to a numeric grade, but not a direct formula.

Much more on this in the [syllabus](#)! (and over the next few weeks)

Minimum Grade	Requirements
<i>Total ESN available</i>	33
3.5	27 Es and 3 additional S+, <u>no Us</u>
3.0	22 Es and 5 additional S+, <u>no Us</u>
2.5	17 Es and 7 additional S+, <u>no Us</u>
2.0	21 S+
1.5	14 S+
0.7	8 S+

S+ = S or E

Agenda: Culture and Community

Today:

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

On Friday:

1. More programming!

Collaboration Policy

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

Some 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 a way not allowed by our [AI policy](#)

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

CSE 121 AI Policy

No part of any graded work may touch an AI tool.

You may not copy and paste any work generated by AI into any graded submission, nor may you copy and paste any work from or for a graded assignment into an AI tool. All other uses of AI on graded work must be cited.

ALLOWED

- Asking AI to **explain an error message**
- Asking AI to **explain the functionality of non-graded code** snippets
- Asking AI to **suggest additional information** or resources

PROHIBITED

- **Generating code, comments, reflections**
- Using AI to **'solve' an assignment**
- Using AI to **write, modify, or extend** reflections, code, comments, etc.

AI and CSE 121: Our Philosophy

Computing applications enabled by **artificial intelligence (AI)** are increasingly common and more widely used for a variety of tasks.

It is becoming more difficult to teach an introductory computing course without acknowledging the **existence of AI tools**.

But as first-time programmers, **you still need to learn and practice effectively using core programming ‘building blocks’**.

This is what CSE 121 is for.

August 2025

41



The Professors Are Using ChatGPT, and Some Students Aren't Happy About It

Students call it hypocritical. A senior at Northeastern University demanded her tuition back. But instructors say generative A.I. tools make them better at their jobs.



Ella Stapleton said she was surprised to find that a professor had used ChatGPT to assemble course materials. "He's telling us not to use it, and then he's using it himself," she said. Oliver Holms for The New York Times



By **Kashmir Hill**

AI & Trust

- we're asking you to **trust** us that not using AI (on graded work) is better for your learning
- in return, we promise you:
 - we will **never** use AI tools to grade your work
 - a human reads, reviews, and provides feedback on **everything** you submit
 - the *only* ways course staff use AI in this class are recording transcriptions, spellcheckers, and [OCR](#)

[The Professors Are Using ChatGPT, and Some Students Aren't Happy About It,](#)

Kashmir Hill, May 14, 2025, The New York Times.

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 Friday, 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](#)