

# CSE 121 – Lesson 19

Miya Natsuhara

Spring 2023

Music: [121 23sp Lecture Vibes](#) 



[sli.do #cse121](#)

TAs:

<i>Jasmine</i>	<i>Atharva</i>	<i>Mia</i>	<i>Justin</i>
<i>Shananda</i>	<i>Julia</i>	<i>Archit</i>	<i>Aishah</i>
<i>Vidhi</i>	<i>Anju</i>	<i>Grace</i>	<i>Claire</i>
<i>Larry</i>	<i>Lydia</i>	<i>Kailye</i>	<i>Lydia</i>
<i>Jacqueline</i>	<i>Jonus</i>	<i>Joshua</i>	<i>Kai</i>
<i>Afifah</i>	<i>Hugh</i>	<i>James</i>	

# Announcements, Reminders

- Gumball (& friends) Visit on Monday, June 5 1:00pm-3:00pm
- Final Exam: **Thursday, June 8 2:30pm-4:20pm**
- Bob Bandes TA Award Nominations Open!
- Course Evaluations open now, and close Sunday June 4 at 11:59pm
  - Currently at about 16% response rate!
- Resubmission Cycle 7 is our *last* resubmission cycle and will close on Thursday, June 8 at 11:59pm
- [Minimum Grade Estimator](#)

# You did it!!



# Learning Objectives

*or, “What did I learn in this class?”*

- 1. Computational Thinking** *Create an algorithm to solve a given problem and express that algorithm in a structured way (e.g. pseudocode)*
- 2. Comprehension** Trace and predict the behavior of programs and systems
- 3. Code Writing** *Write functionally correct Java programs that meet a provided specification using control structures, primitive data types, and basic data abstractions*
- 4. Communication** *Clearly and effectively describe the behavior of a given code snippet*
- 5. Debugging** *Identify errors in a method’s behavior & implement fixes for identified errors*
- 6. Decomposition** Solve problems by breaking them into subproblems and recombining the solutions using techniques such as methods
- 7. Ethics/Impact** *Describe ethical and sociotechnical issues related to software and technology and explain how their choices as programmers can impact those issues*

# (Optional) Quines

*A quine is a computer program which takes no input and produces a copy of its own source code as its only output.*

# (Optional) Quines

A quine in Java!

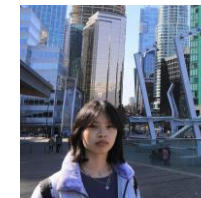
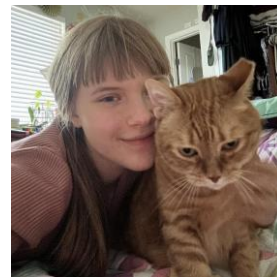
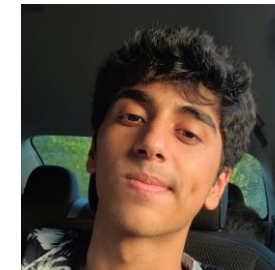
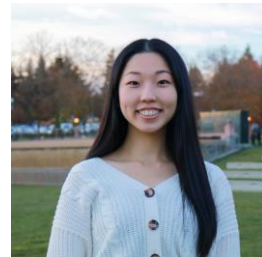
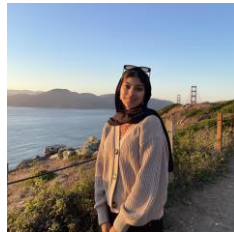
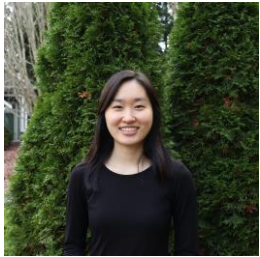
You can understand all of the code constructs and syntax used here!

(Even if you would never have written code like this yourself)

```
public class Quine {
    public static void main(String[] args){
        char q = 34;        // Quotation mark character
        String[] l = {      // Array of source code
            "public class Quine",
            "{",
            "    public static void main(String[] args)",
            "    {",
            "        char q = 34;        // Quotation mark character",
            "        String[] l = {      // Array of source code",
            "            ",
            "        };",
            "        for (int i = 0; i < 6; i++)          // Print opening code",
            "            System.out.println(l[i]);",
            "        for (int i = 0; i < l.length; i++)      // Print string array",
            "            System.out.println(l[6] + q + l[i] + q + ',');",
            "        for (int i = 7; i < l.length; i++)      // Print this code",
            "            System.out.println(l[i]);",
            "    }",
            "}",
        };
        for (int i = 0; i < 6; i++)          // Print opening code
            System.out.println(l[i]);
        for (int i = 0; i < l.length; i++)    // Print string array
            System.out.println(l[6] + q + l[i] + q + ',');
        for (int i = 7; i < l.length; i++)    // Print this code
            System.out.println(l[i]);
    }
}
```



# Thank your TAs!



# Thank you!

- This is still a very new course! We are always looking for feedback on how to improve the class for you and for future students! Thank you for your patience and understanding as we continue to improve these new assignments, resources, and examples.
  - *We really* value your feedback!
  - Let us know what is or isn't working for you!
  - Something that went well in another course? Tell us about it!
- ...Please fill out course evals by **Sunday June 4 at 11:59pm** to provide feedback about the course!



# Applications of CS

*or “What can I do with what I learned?”*

- *Detect and prevent toxicity online*
- *Digitize basketball players*
- *Help DHH people identify sounds*
- *Figure out how to best distribute relief funds*
- *Recognize disinformation online*
- *Make movies*
- *Improve digital collaboration*
- *Fix Olympic badminton* & *Identintify cheating in chess*
- *And so much more!*

# Future Courses

*or “What can I do next?”*

Course	Overview
<a href="#">CSE 122</a>	Introduction to Computer Programming II
<a href="#">CSE 123</a>	Introduction to Computer Programming III

## Majors

Course	Overview
<a href="#">CSE 311</a>	Mathematical foundations
<a href="#">CSE 351</a>	Low-level computer organization/abstraction
<a href="#">CSE 331</a>	Software design/implementation
<a href="#">CSE 341</a>	Programming languages
<a href="#">CSE 340</a>	Interaction programming

## Non-majors

Course	Overview
<a href="#">CSE 154</a>	Intro. to web programming (several languages)
<a href="#">CSE 163</a>	Intermediate programming, data analysis (Python)
<a href="#">CSE 180</a>	Introduction to data science (Python)
<a href="#">CSE 373</a>	Data structures and algorithms (non-majors)
<a href="#">CSE 374</a>	Low-level programming and tools (C/C++)
<a href="#">CSE 416</a>	Intro. to Machine Learning

Other tech-related majors:

Informatics, ACMS, Electrical & Computer Engineering, ...

See: <https://www.cs.washington.edu/academics/ugrad/current-students> and <https://www.cs.washington.edu/academics/ugrad/nonmajor-options/nonmajor-courses>

# Frequently Asked Questions

- *How can I get better at programming?*
  - Practice!
- *How can I learn to X?*
  - Search online, read books, look at examples :)
- *What should I work on next?*
  - Anything you can think of! ([Here are some ideas](#))
  - Beware: it's hard to tell what's easy and what's hard.
- *Should I learn another language? Which one?*
  - That depends—what do you want to do?
- *What's the best programming language?*
  - 😞 (take CSE 341 or CSE 413)

# Thank you!

Ask Me (Almost)  
Anything!

