CSE 121 – Lesson 14

Kai Daniels

Summer 2023

Music: 🌸k-pop girlies playlist🌸

No sli.do today <3

Announcements, Reminders

- Creative Project 3 Practice Final released, due next Tuesday
- Resub 5 due tomorrow, Resub 6 (final resub) out tomorrow
- Course Evaluations open Aug 12th – Aug 18th
  - https://uw.iasystem.org/survey/277506
- **Reminder:** Final exam Wednesday Aug 16 4:30 – 6:30 PM in PAA A102
  - See Final Exam page for the procedures and seating Assignments
  - Friday Lecture is Final Review Session

(PCM) Looking at Multiple Elements in an Array

0	1	9	1	0

```
public static boolean isPalindrome(int[] list) {
    for (int i = 0; i < list.length / 2; i++) {
        if (list[i] != list[list.length - 1 - i]) {
            return            ;
        }
    }

    return           ;
}
```

(PCM) Looking at Multiple Elements in an Array

0	1	9	1	0

```
public static boolean isPalindrome(int[] list) {
    for (int i = 0; i < list.length / 2; i++) {
        if (list[i] != list[list.length - 1 - i]) {
            return false;
        }
    }

    return true;
}
```

(PCM) Array of Counters or "Tallying"

```
8 3 0 1 2 2 0 7 2
```

```
public static int[] numCount(Scanner input) {
    int[] counts =              ;
    while (input.hasNextInt()) {
        int num = input.nextInt();



    }

    return counts;
}
```

(PCM) Array of Counters or "Tallying"

```
8 3 0 1 2 2 0 7 2
```

```
public static int[] numCount(Scanner input) {
    int[] counts = new int[10];
    while (input.hasNextInt()) {
        int num = input.nextInt();
        counts[num]++;
    }

    return counts;
}
```

(PCM) Common Ideas in Array Patterns

- Loop bounds
- Direction of traversal
- Indexing into an array

Final exam

- To C3 and the website!

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*

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 **August 18th at 11:59pm** to provide feedback about the course!

Future Courses

or "What can I do next?"

Course	Overview
CSE 122	Introduction to Computer Programming II
CSE 123	Introduction to Computer Programming III

Majors

Course	Overview
CSE 311	Mathematical foundations
CSE 351	Low-level computer organization/abstraction 😉
CSE 331	Software design/implementation
CSE 341	Programming languages
CSE 340	Interaction programming

Non-majors

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

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!*

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)