

CSE 121 – Lesson 18

Miya Natsuhara

Spring 2023

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



[sli.do #cse121](https://sli.do/#cse121)

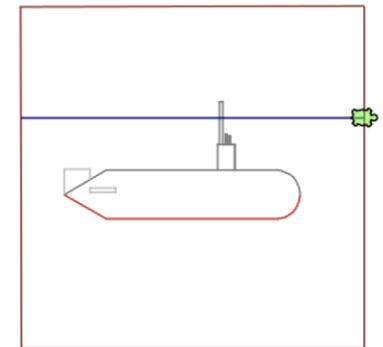
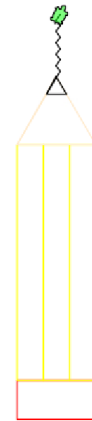
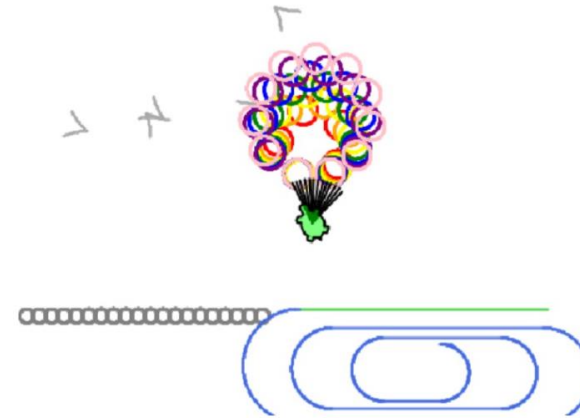
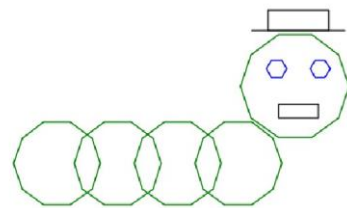
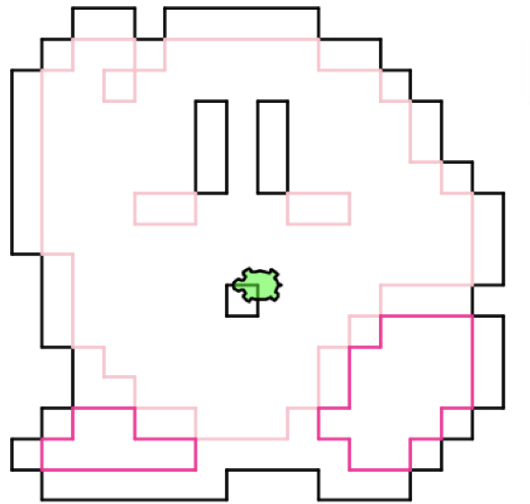
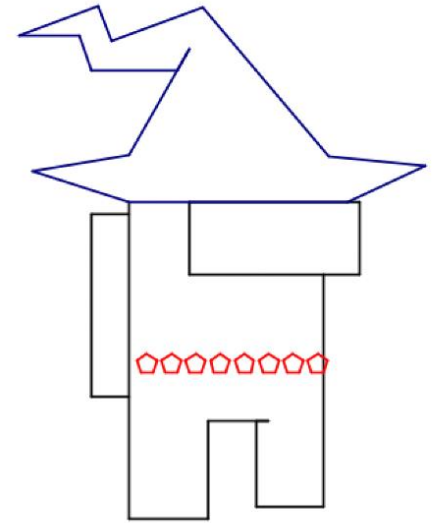
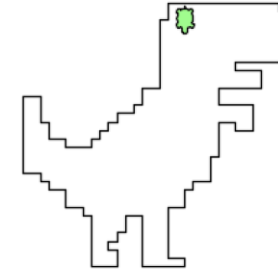
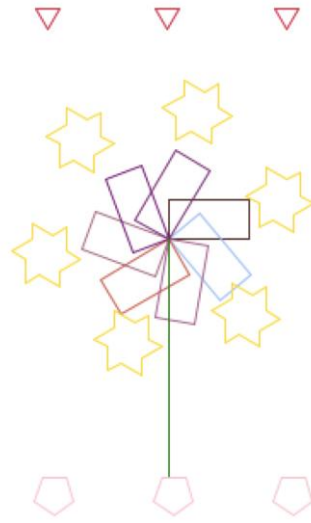
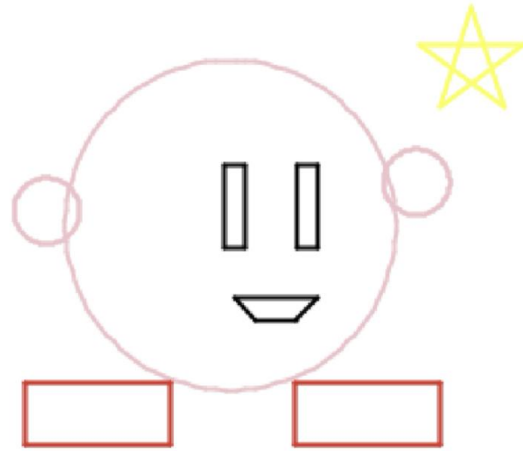
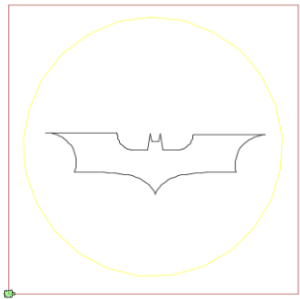
TAs:

<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

- P3 due **tonight** Wednesday, May 31 11:59pm
- Gumball (& friends) Visit on Monday, June 5 1:00pm-3:00pm
- Final Exam: **Thursday, June 8 2:30pm-4:20pm**
 - TA-led Final Review Session Tuesday, June 6 4:30pm-7:00pm
 - One 8.5x11 inch sheet of notes (double-sided, handwritten or typed)
 - Seating assignments posted!
- Bob Bandes TA Award Nominations Open!
- Course Evaluations open now, and close Sunday June 4 at 11:59pm
 - Currently at about 11% response rate!
- Also do your TAs' evals in section tomorrow!

C3 Drawings!



Poll in with your answer!



```
public static void main(String[] args) {
    int x = 0;
    int[] a = new int[4];
    x++;

    mystery(x, a);
    System.out.println(x + " " + Arrays.toString(a));

    x++;
    mystery(x, a);
    System.out.println(x + " " + Arrays.toString(a));
}
```

```
public static void mystery(int x, int[] a) {
    x++;
    a[x]++;
    System.out.println(x + " " + Arrays.toString(a));
}
```

Four lines of output would be produced by this code. What would those four lines be?

Poll in with your answer!



```
public static void main(String[] args) {
    int x = 0;
    int[] a = new int[4];
    x++;

    mystery(x, a);
    System.out.println(x + " " + Arrays.toString(a));

    x++;
    mystery(x, a);
    System.out.println(x + " " + Arrays.toString(a));
}

public static void mystery(int x, int[] a) {
    x++;
    a[x]++;
    System.out.println(x + " " + Arrays.toString(a));
}
```

(PCM) Counting Elements that Meet a Condition

"one"	"two"	"three"	"six"	"seven"	"eight"	"ten"
-------	-------	---------	-------	---------	---------	-------

```
public static int evenLength(String[] list) {  
    int countEven = 0;  
    for (int i = 0; i < list.length; i++) {  
        if (                ) {  
            countEven++;  
        }  
    }  
  
    return countEven;  
}
```

(PCM) Modifying Elements of an Array

4	8	15	16	23	42
---	---	----	----	----	----

```
public static void clamp(int min, int max, int[] list) {  
    for (int i = 0; i < list.length; i++) {  
        if (                > max) {  
                = max;  
        } else if (                < min) {  
                = min;  
        }  
    }  
}
```

(PCM) Searching for an Element

"one"	"two"	"three"	"six"	"seven"	"eight"	"ten"
-------	-------	---------	-------	---------	---------	-------

```
public static int indexOfIgnoreCase(String phrase, String[] list) {  
    for (int i = 0; i < list.length; i++) {  
        if (                ) {  
            return          ;  
        }  
    }  
  
    return          ;  
}
```


(PCM) Shifting Elements

9.6	-88.0	4.815	0.009	7.0184	42.9
-----	-------	-------	-------	--------	------

```
public static void rotateRight(double[] list) {  
    double lastElement = list[list.length - 1];  
  
    for (int i = list.length - 1; i > 0; i--) {  
        list[i] = list[i - 1];  
    }  
  
}
```

(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) Common Ideas in Array Patterns

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