

# CSE 121 – Lesson 16

Elba Garza & Matt Wang

Winter 2024



TAs: Abby    Aishah    Anju    Annie    Archit    Ayesha    Christian  
Hannah    Heather    Hibbah    Jacob    James    Janvi    Jasmine  
Jonus    Julia    Lucas    Luke    Maria    Nicole    Shananda  
Shayna    Trey    Vidhi    Vivian

Today's playlist:  
[CSE 121 24wi lecture beats :D](#)

[sli.do #cse121-16](https://sli.do/#cse121-16)

# Announcements, Reminders

- Programming Assignment 3 releasing later today – last assignment!
  - Focused on 2D arrays!
  - No dedicated resubmission cycle  
(Feedback will not be released the following Tuesday)
  - Extending P3's planned deadline to **Thursday** next week (March 7th)...  
*so you have an additional 2 days to work on it!*
- Gigi (& friends) Visit on Monday, March 11th 1:00pm-3:00pm
- Final Exam: **Tuesday, March 12th 12:30pm-2:20pm**
  - [Left-Handed Seating Requests Form](#), closes end-of-day Monday, March 4<sup>th</sup>

# Final Exam Details

- Final Exam: **Tuesday, March 12th 12:30pm-2:20pm**
- In-person, on paper, with assigned seating
- No collaboration – should be completed individually
- You may bring in **one 8.5x11-inch** sheet of notes, handwritten or typed, double-sided
- Focused on behavior (not style)
- You should make every attempt to write correct Java syntax
- Next week will be focused on Final Exam review and preparation
  - We are also planning a Final Exam Review Session on Monday of finals week (March 11<sup>th</sup>)
- More details soon on course website!

# 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) 2D Arrays

```
int[][] a = new int[4][3];
```

type            name            array creation code

*An array of arrays!*

- The *ElementType* of the array is another array itself!
  - Your first example of “nested data structures”
    - There will be more in CSE 122!

int[][]

double[][]

String[][]

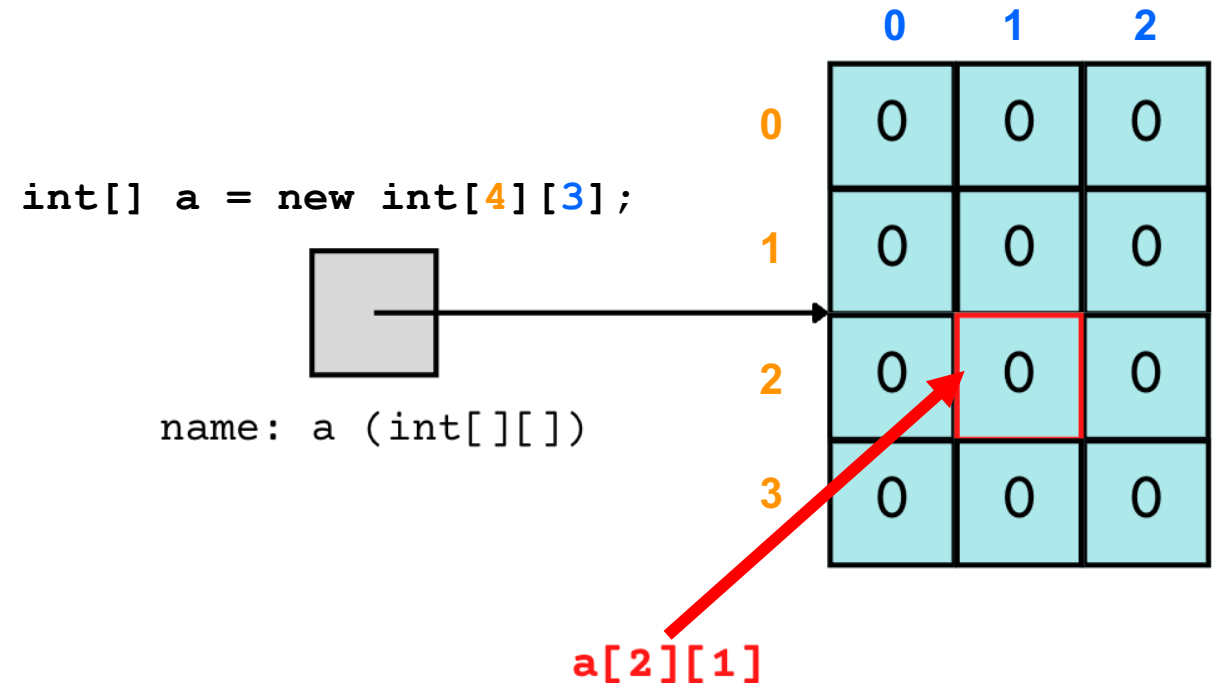
boolean[][]

char[][]

# (PCM) 2D Arrays

*An array of arrays!*

The two dimensions are  
“rows” and “columns”

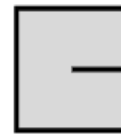


# (PCM) 2D Arrays

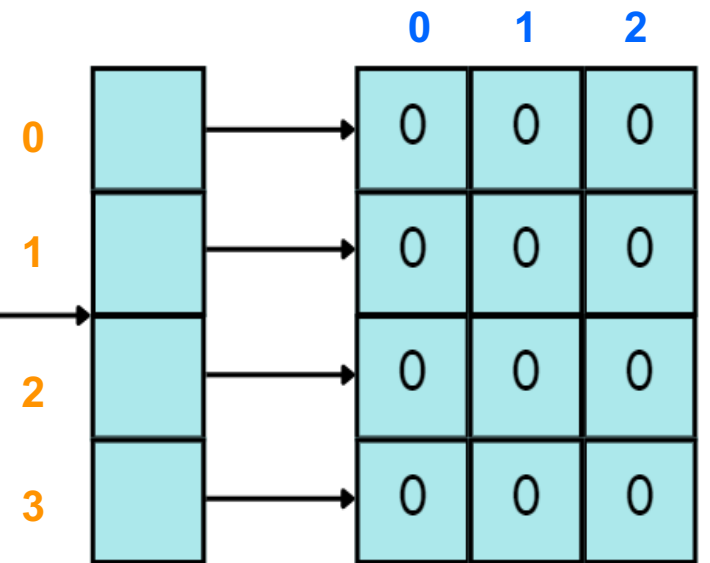
A slightly more accurate view...

*reference semantics*

```
int[] a = new int[4][3];
```



name: a (int[][])





# (PCM) 2D Array Traversals

*for each row...*

```
for (int i = 0; i < list.length; i++) {  
    for (int j = 0; j < list[i].length; j++) {  
        // do something with list[i][j]  
    }  
}
```

*for each element within a row...*

# Arrays Utility Class

Method	Description
<code>Arrays.toString(array);</code>	Returns a <code>String</code> representing the array, such as <code>"[10, 30, -25, 17]"</code>
<code>Arrays.fill(array, value);</code>	Sets every element to the given value
<code>Arrays.equals(array1, array2);</code>	Returns true if the two arrays contain the same elements in the same order
<code>Arrays.deepToString(array);</code>	Returns a <code>String</code> representing the array; if the array contains other arrays as elements, the <code>String</code> represents their contents, and so on. For example, <code>"[[99, 151], [30, 5]]"</code>
<code>Arrays.deepEquals(array1, array2);</code>	Returns true if the two arrays contain the same elements in the same order; if the array(s) contain other arrays as elements, their contents are tested for equality, and so on.

# Applications of 2D Arrays

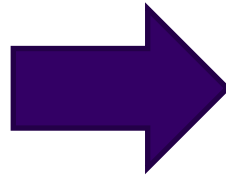
- Matrices
  - Useful in various applications requiring complex math!
- Board games
  - (e.g., chess/checkerboard, tic tac toe, sudoku)
- Representing information in a grid or table
  - (e.g., scorekeeping, gradebook)
- Image processing

# matrixAdd

23	96	18	4	64
45	40	18	44	34
92	13	77	71	12



70	73	66	79	39
91	75	73	99	47
27	64	21	34	1




# matrixAdd

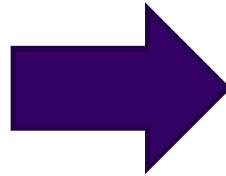
23	96	18	4	64
45	40	18	44	34
92	13	77	71	12



70	73	66	79	39
91	75	73	99	47
27	64	21	34	1

i: 0

j: 0



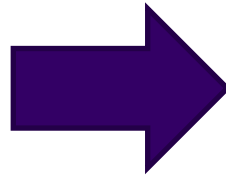

# matrixAdd

23	96	18	4	64
45	40	18	44	34
92	13	77	71	12



70	73	66	79	39
91	75	73	99	47
27	64	21	34	1

i: 0  
j: 0



93				

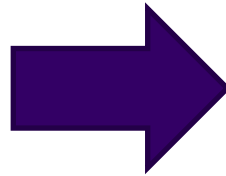
# matrixAdd

23	96	18	4	64
45	40	18	44	34
92	13	77	71	12



70	73	66	79	39
91	75	73	99	47
27	64	21	34	1

i: 0  
j: 1



93	169			

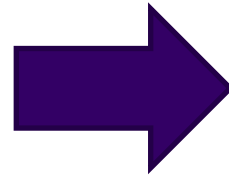
# matrixAdd

23	96	18	4	64
45	40	18	44	34
92	13	77	71	12



70	73	66	79	39
91	75	73	99	47
27	64	21	34	1

i: 0  
j: 2



93	169	84		



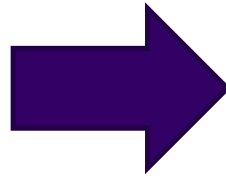
# matrixAdd

23	96	18	4	64
45	40	18	44	34
92	13	77	71	12



70	73	66	79	39
91	75	73	99	47
27	64	21	34	1

i: 0  
j: 3



93	169	84	83	

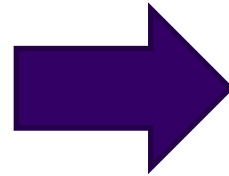
# matrixAdd

23	96	18	4	64
45	40	18	44	34
92	13	77	71	12



70	73	66	79	39
91	75	73	99	47
27	64	21	34	1

i: 0  
j: 4



93	169	84	83	103

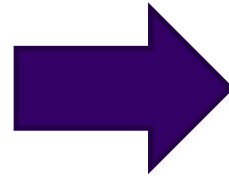
# matrixAdd

23	96	18	4	64
45	40	18	44	34
92	13	77	71	12



70	73	66	79	39
91	75	73	99	47
27	64	21	34	1

i: 1  
j: 0



93	169	84	83	103
136				

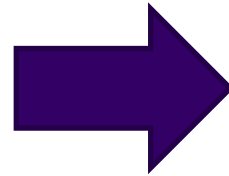
# matrixAdd

23	96	18	4	64
45	40	18	44	34
92	13	77	71	12



70	73	66	79	39
91	75	73	99	47
27	64	21	34	1

i: 1  
j: 1



93	169	84	83	103
136	115			

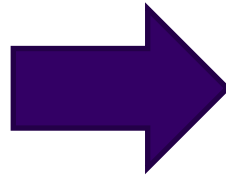
# matrixAdd

23	96	18	4	64
45	40	18	44	34
92	13	77	71	12



70	73	66	79	39
91	75	73	99	47
27	64	21	34	1

i: 1  
j: 2



93	169	84	83	103
136	115	91		

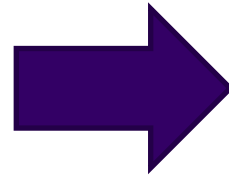
# matrixAdd

23	96	18	4	64
45	40	18	44	34
92	13	77	71	12



70	73	66	79	39
91	75	73	99	47
27	64	21	34	1

i: 1  
j: 3



93	169	84	83	103
136	115	91	143	

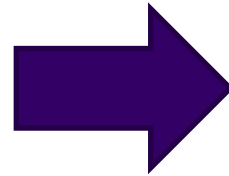
# matrixAdd

23	96	18	4	64
45	40	18	44	34
92	13	77	71	12



70	73	66	79	39
91	75	73	99	47
27	64	21	34	1

i: 1  
j: 4



93	169	84	83	103
136	115	91	143	81

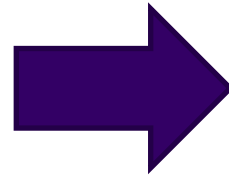
# matrixAdd

23	96	18	4	64
45	40	18	44	34
92	13	77	71	12



70	73	66	79	39
91	75	73	99	47
27	64	21	34	1

i: 2  
j: 0



93	169	84	83	103
136	115	91	143	81
119				



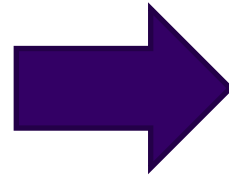
# matrixAdd

23	96	18	4	64
45	40	18	44	34
92	13	77	71	12



70	73	66	79	39
91	75	73	99	47
27	64	21	34	1

i: 2  
j: 1



93	169	84	83	103
136	115	91	143	81
119	77			

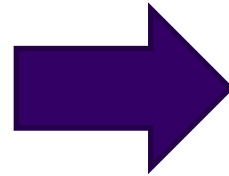
# matrixAdd

23	96	18	4	64
45	40	18	44	34
92	13	77	71	12



70	73	66	79	39
91	75	73	99	47
27	64	21	34	1

i: 2  
j: 2



93	169	84	83	103
136	115	91	143	81
119	77	98		

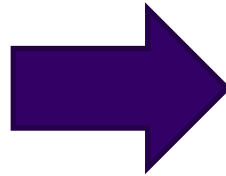
# matrixAdd

23	96	18	4	64
45	40	18	44	34
92	13	77	71	12



70	73	66	79	39
91	75	73	99	47
27	64	21	34	1

i: 2  
j: 3



93	169	84	83	103
136	115	91	143	81
119	77	98	105	

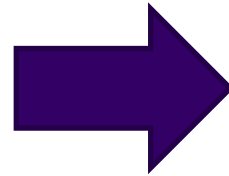
# matrixAdd

23	96	18	4	64
45	40	18	44	34
92	13	77	71	12



70	73	66	79	39
91	75	73	99	47
27	64	21	34	1

i: 2  
j: 4



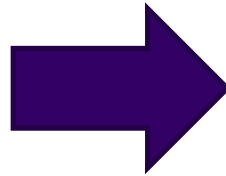
93	169	84	83	103
136	115	91	143	81
119	77	98	105	13

# matrixAdd

23	96	18	4	64
45	40	18	44	34
92	13	77	71	12



70	73	66	79	39
91	75	73	99	47
27	64	21	34	1



93	169	84	83	103
136	115	91	143	81
119	77	98	105	13

## (2D)ays Above Average: `readData()`

How many days' data would you like to input? 3

Next day's data:

Temperature in Seattle? 44

Temperature in Tacoma? 40

Temperature in Bothell? 43

Next day's data:

Temperature in Seattle? 42

Temperature in Tacoma? 40

Temperature in Bothell? 44

Next day's data:

Temperature in Seattle? 42

Temperature in Tacoma? 41

Temperature in Bothell? 43

...



	Seattle	Tacoma	Bothell
1	44	40	43
2	42	40	44
3	42	41	43

## (2D)ays Above Average: `readData()`

How many days' data would you like to input? 3

Next day's data:

Temperature in Seattle? 44

Temperature in Tacoma? 40

Temperature in Bothell? 43

Next day's data:

Temperature in Seattle? 42

Temperature in Tacoma? 40

Temperature in Bothell? 44

Next day's data:

Temperature in Seattle? 42

Temperature in Tacoma? 41

Temperature in Bothell? 43

...



	Seattle	Tacoma	Bothell
1	44		
2			
3			

## (2D)ays Above Average: `readData()`

How many days' data would you like to input? 3

Next day's data:

Temperature in Seattle? 44

Temperature in Tacoma? 40

Temperature in Bothell? 43

Next day's data:

Temperature in Seattle? 42

Temperature in Tacoma? 40

Temperature in Bothell? 44

Next day's data:

Temperature in Seattle? 42

Temperature in Tacoma? 41

Temperature in Bothell? 43

...



	Seattle	Tacoma	Bothell
1	44	40	
2			
3			



## (2D)ays Above Average: `readData()`

How many days' data would you like to input? 3

Next day's data:

Temperature in Seattle? 44

Temperature in Tacoma? 40

Temperature in Bothell? 43

Next day's data:

Temperature in Seattle? 42

Temperature in Tacoma? 40

Temperature in Bothell? 44

Next day's data:

Temperature in Seattle? 42

Temperature in Tacoma? 41

Temperature in Bothell? 43

...



	Seattle	Tacoma	Bothell
1	44	40	43
2			
3			

## (2D)ays Above Average: `readData()`

How many days' data would you like to input? 3

Next day's data:

Temperature in Seattle? 44

Temperature in Tacoma? 40

Temperature in Bothell? 43

Next day's data:

Temperature in Seattle? 42

Temperature in Tacoma? 40

Temperature in Bothell? 44

Next day's data:

Temperature in Seattle? 42

Temperature in Tacoma? 41

Temperature in Bothell? 43

...



	Seattle	Tacoma	Bothell
1	44	40	43
2	42		
3			

## (2D)ays Above Average: `readData()`

How many days' data would you like to input? 3

Next day's data:

Temperature in Seattle? 44

Temperature in Tacoma? 40

Temperature in Bothell? 43

Next day's data:

Temperature in Seattle? 42

Temperature in Tacoma? 40

Temperature in Bothell? 44

Next day's data:

Temperature in Seattle? 42

Temperature in Tacoma? 41

Temperature in Bothell? 43

...



	Seattle	Tacoma	Bothell
1	44	40	43
2	42	40	
3			

## (2D)ays Above Average: `readData()`

How many days' data would you like to input? 3

Next day's data:

Temperature in Seattle? 44

Temperature in Tacoma? 40

Temperature in Bothell? 43

Next day's data:

Temperature in Seattle? 42

Temperature in Tacoma? 40

Temperature in Bothell? 44

Next day's data:

Temperature in Seattle? 42

Temperature in Tacoma? 41

Temperature in Bothell? 43

...



	Seattle	Tacoma	Bothell
1	44	40	43
2	42	40	44
3			

## (2D)ays Above Average: `readData()`

How many days' data would you like to input? 3

Next day's data:

Temperature in Seattle? 44

Temperature in Tacoma? 40

Temperature in Bothell? 43

Next day's data:

Temperature in Seattle? 42

Temperature in Tacoma? 40

Temperature in Bothell? 44

Next day's data:

Temperature in Seattle? 42

Temperature in Tacoma? 41

Temperature in Bothell? 43

...



	Seattle	Tacoma	Bothell
1	44	40	43
2	42	40	44
3	42		

## (2D)ays Above Average: `readData()`

How many days' data would you like to input? 3

Next day's data:

Temperature in Seattle? 44

Temperature in Tacoma? 40

Temperature in Bothell? 43

Next day's data:

Temperature in Seattle? 42

Temperature in Tacoma? 40

Temperature in Bothell? 44

Next day's data:

Temperature in Seattle? 42

Temperature in Tacoma? 41

Temperature in Bothell? 43

...



	Seattle	Tacoma	Bothell
1	44	40	43
2	42	40	44
3	42	41	

## (2D)ays Above Average: `readData()`

How many days' data would you like to input? 3

Next day's data:

Temperature in Seattle? 44

Temperature in Tacoma? 40

Temperature in Bothell? 43

Next day's data:

Temperature in Seattle? 42

Temperature in Tacoma? 40

Temperature in Bothell? 44

Next day's data:

Temperature in Seattle? 42

Temperature in Tacoma? 41

Temperature in Bothell? 43

...



	Seattle	Tacoma	Bothell
1	44	40	43
2	42	40	44
3	42	41	43

# (2D)ays Above Average: `computeAverages()`

How many days' data would you like to input? 3

...

The average values for each location were  
[42.666666666666664, 40.333333333333336,  
43.333333333333336]

	Seattle	Tacoma	Bothell
1	44	40	43
2	42	40	44
3	42	41	43



42.667	40.333	43.333
--------	--------	--------

Average of Seattle  
temperatures  
 $(44 + 42 + 42) / 3$