

Lec 25

Conclusion

```
public class Test {  
    public static int i = 0;  
  
    public static void main(String[] args) {  
        Test t = new Test(i++);  
        String s = "hi";  
        t.method1(i, s);  
        System.out.println(i + s);  
    }  
  
    public Test(int i) {  
        this.i = i*2;  
    }  
  
    public void method1(int i, String s) {  
        this.i = i++;  
        s += this.i;  
    }  
}
```

- Turn in extra credit by this Friday.
- Projects due midnight March 12th
 - No slip days, no extensions, no excuses
- Project demos on March 13th, 1-3
 - Bring your laptop with a charged battery, format will be like a science fair where everyone can walk around and see your project
- Next class is optional. Normal class location.
- 8AM: Final exam in this room, Monday 8-11am
- 2PM: Final exam in this room, Monday 3-6pm

We covered

- Linux
- Hello world
- Binary
- Algorithms
- Run time
- Growth
- Big O
- Editors
- Comments
- Style
- Scope
- Variables
- Primitive types
- Conditionals
- Order of operations
- Expressions
- Compound operators
- Turtles
- Casting
- Loops
- Switch statements
- Methods
- Method overloading
- User input
- Parameters
- Arrays
- Memory
- File IO
- Buffering
- ArrayLists
- Sorting
- Searching
- HashSet
- HashMap
- Classes
- UML
- Access modifiers
- Constructors
- *this*
- *super*
- Stack
- Heap
- Static
- Junit tests
- Threads
- Inheritance
- Polymorphism
- Declared type
- Actual type
- Method overriding
- Abstract classes
- Interfaces
- IDEs
- JavaFX
- GUIs
- Events
- Exceptions
- Animations
- Recursion
- Candy crush
- GTA

Stories on

- Importance on getting involved
- Changing car tires
- Pirates of the Caribbean
- Middle school and order of operations
- Blind dates
- Kanye Tweets
- Deciding how many kids you want to have and buying a car
- Amsterdam and HashSets
- Rouge programming
- Pokémon programming
- Dragons

Looking ahead... life after CSE 11?

What was your favorite/most compelling idea from CSE11?

Do you want to know more about how to store and find data efficiently?

HW6

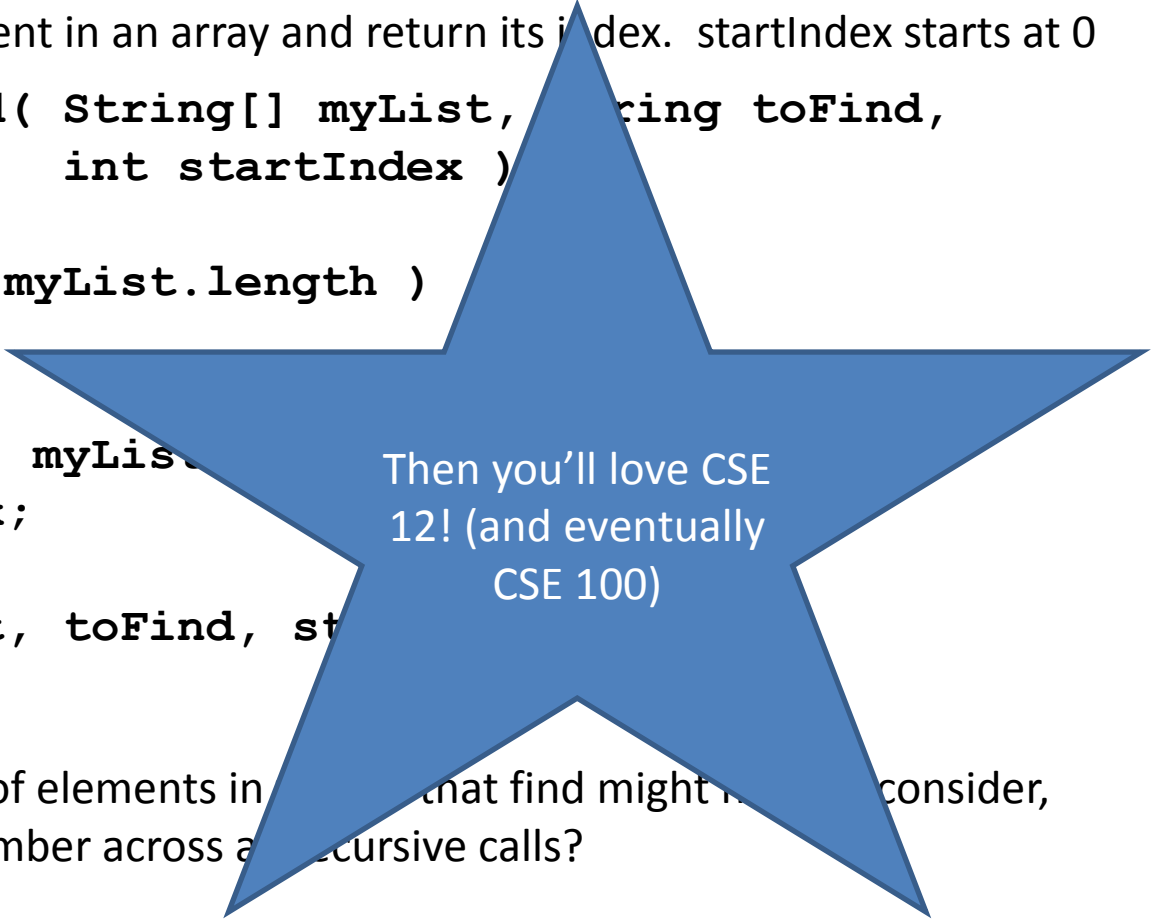
Collections

- ArrayList
- HashSet/Map

If you want to know more about how to store and find data efficiently...

Write a method to find an element in an array and return its index. `startIndex` starts at 0

```
public static int find( String[] myList, String toFind,
                      int startIndex )
{
    if ( startIndex >= myList.length )
        return -1;
    }
    if ( toFind.equals( myList[startIndex] ) )
        return startIndex;
    }
    return find( myList, toFind, startIndex + 1 );
}
```



Then you'll love CSE
12! (and eventually
CSE 100)

What is the **maximum** number of elements in `myList` that `find` might have to consider, taking into account the *total* number across all recursive calls?

Do you want to know more about how to debug?

HW7

- Unit testing
- Debugging
- IDEs (Eclipse)

If you ever had a bug you couldn't find...

"For some reason my critters only move north... why??"

"My 2048 code always thinks it can move right, even when it shouldn't... why??"

"I'm getting an `ArrayIndexOutOfBoundsException` error. Why??"



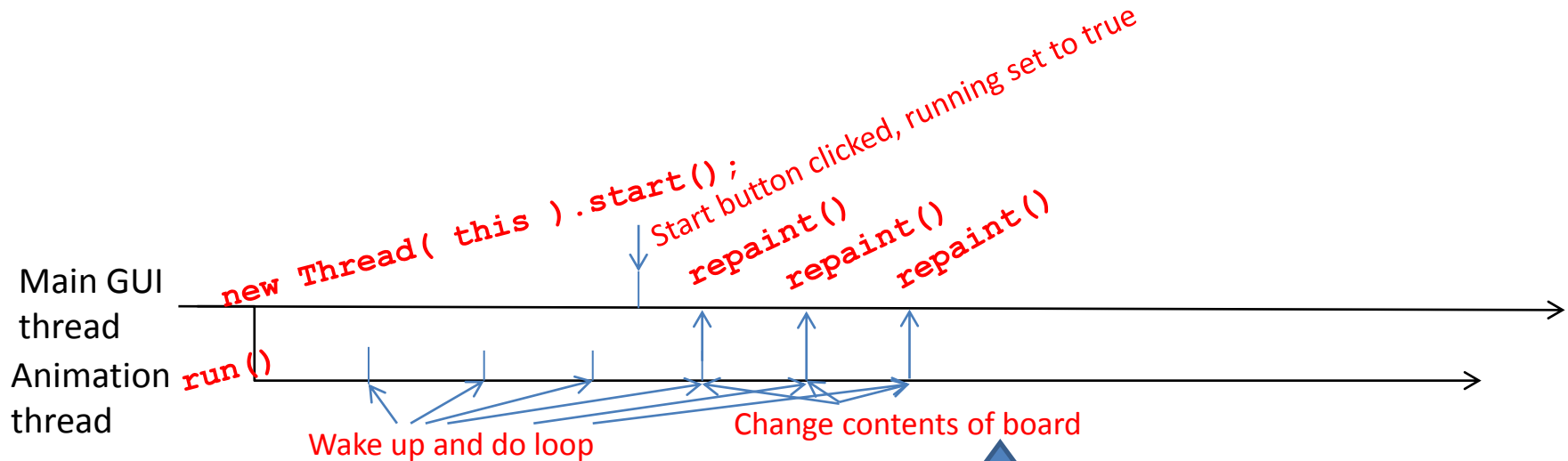
Then you'll
love CSE 15L!

Did you wonder how your computer runs so many things at the same time and how it manages hardware?

HW8

- Threads

Did you wonder how your computer runs so many things at the same time and how it manages hardware?



Then you'll love
CSE 120!

Do you want to learn how to build larger programs?

HW9

Final Project

How to design large programs

Work in teams

If you liked building larger programs...



Then you'll love
CSE 110!

And so much more!

Of these, which was your favorite PA?

- A. PA4: poker
- B. PA5: File recovery
- C. PA6: WordCloud
- D. PA7: 2048+testing
- E. PA8: ObjectDraw

Of these, which was your favorite PA?

- A. Favorite last slide
- B. Second favorite last slide
- C. PA9: Business program w/ excel
- D. Final Project

Trace/debug: interpreting Java errors

In what method that you wrote does the error occur?

A. main

 B. sum

C. main and sum

D. neither

```
java.lang.ArrayIndexOutOfBoundsException: 5
```

```
...
```

```
at ArrayPlayReview.sum(ArrayPlayReview.java:7)
```

```
at ArrayPlayReview.main(ArrayPlayReview.java:15)
```

```

public class Weapon {
    public Weapon() {
        System.out.println("weapon: default");
    }

    public Weapon(String name) {
        System.out.println("weapon: " + name);
    }
    public void fire() {
        System.out.println("BANG!");
    }
}

public class Gun extends Weapon {
    public Gun() {
        System.out.println("gun: default");
    }

    public Gun(String name) {
        super(name);
        System.out.println("gun: " + name);
    }
}

public class Pistol extends Gun {
    public Pistol() {
        System.out.println("pistol: default");
    }

    public Pistol(String name) {
        System.out.println("pistol: " + name);
    }
    public void fire() {
        System.out.println("pew! pew!");
    }
}

```

```

Weapon ppk = new Pistol("Walther PPK");
ppk.fire();

```

B.

```

Weapon gun = new Gun("Barrett 50");
gun.fire();

```

2. What is the Big-O complexity for each of the following code segments?

```
int[] array = {...};  
for (i = 0; i < array.length; i++) {  
    for (j = i; j < array.length; j += 1) {  
        if (array[i] == array[j])  
            System.out.println("duplicate");  
    }  
}
```

```
for (i = 0; i < n; i++) {  
    for (j = i - 10; j < i + 10; j++) {  
        if (i == j)  
            System.out.println("match");  
    }  
}
```

```
for(i = 0; i < n*n; i++) {  
    for(j = 0; j < i; j++) {  
        if(i == j) {  
            System.out.println("match");  
        }  
    }  
}
```

3. Draw the order of methods on the stack up to the indicated point of program execution, *method*. Note: the stack may not fill completely, also Stacks fill from the bottom to the top:

UCSD's goals for this course

- Write
 - Programs/algorithms
 - Using loops, arrays, conditionals, variables objects, subclasses, GUIs, recursion, etc
- Trace/debug
 - Draw memory models
 - Given some code, what does it do?
 - Find the errors and fix them
- Explain
 - Why things go wrong
 - How things work/why code functions the way it does
 - Why the answer to a question is right or wrong
- HW9 had almost everything that we covered in the course...
 - Inheritance, loops, collections, classes, new APIs, ...

Common interview question

Write a program that prints the numbers from 1 to 100.

*But for multiples of three print “Fizz” instead of the number
and for the multiples of five print “Buzz”.*

For numbers which are multiples of both three and five print “FizzBuzz”

- Can you do it using recursion?
- Can you draw the memory models for both solutions?
- Can you implement without the % operator?

My additional goals for this course

- Independence
- Program Clarity/Documentation/Style
- Mentorship with tutors

Final Thoughts

<http://www.cape.ucsd.edu/>