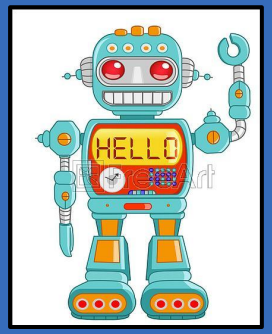


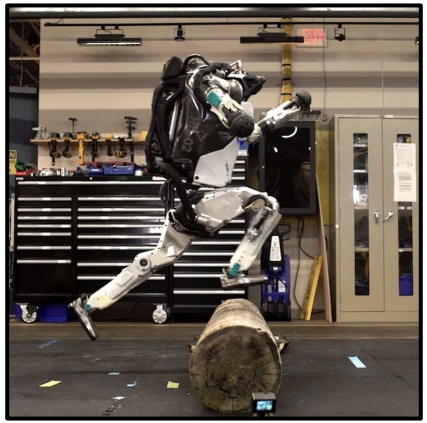
Welcome to CS200!

Sam Vinitsky

CS200 Summer 2019

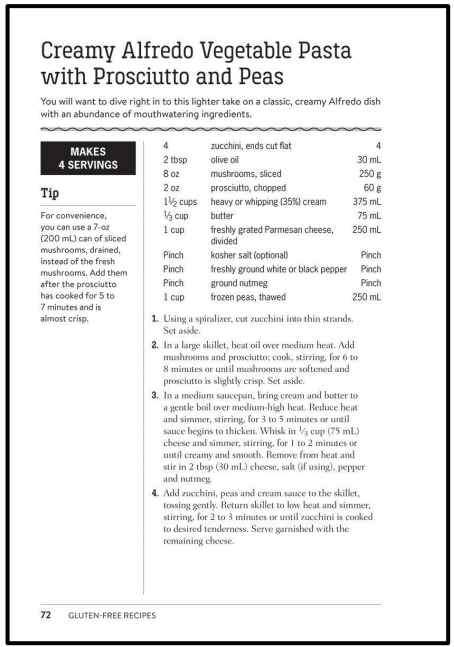
What is this course about?

- Making **computers** do cool things



How can we tell computers what to do?

- **Program:** (or *algorithm*)
  - List of instructions for **accomplishing a task**



Creamy Alfredo Vegetable Pasta with Prosciutto and Peas

You will want to dive right in to this lighter take on a classic, creamy Alfredo dish with an abundance of mouthwatering ingredients.

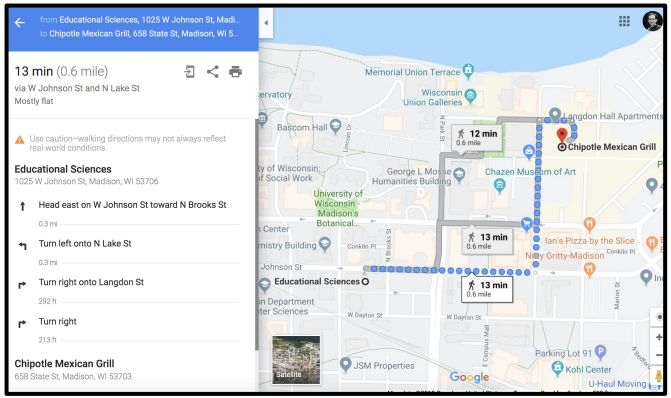
MAKES 4 SERVINGS

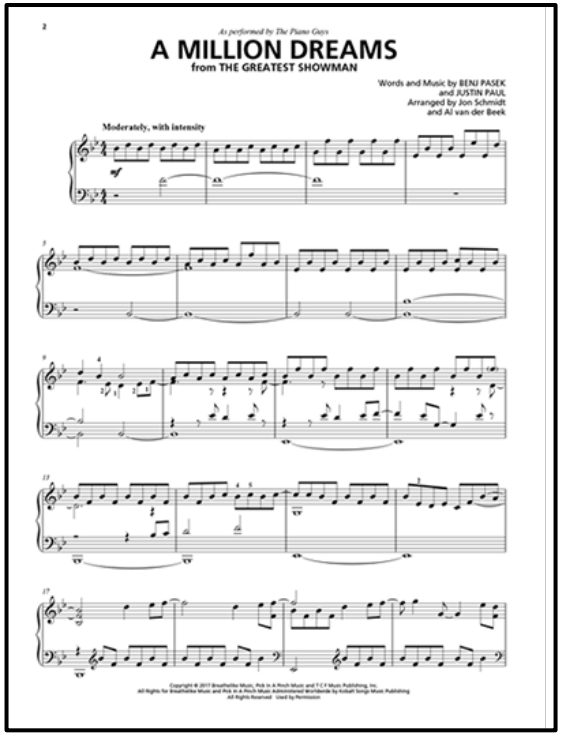
4	zucchini, ends cut flat	4
2 tbsp	olive oil	30 mL
8 oz	mushrooms, sliced	250 g
2 oz	prosciutto, chopped	60 g
1½ cups	heavy or whipping (35%) cream	375 mL
⅓ cup	butter	75 mL
1 cup	freshly grated Parmesan cheese, divided	250 mL
Pinch	kosher salt (optional)	Pinch
Pinch	freshly ground white or black pepper	Pinch
Pinch	ground nutmeg	Pinch
1 cup	frozen peas, thawed	250 mL

Tip

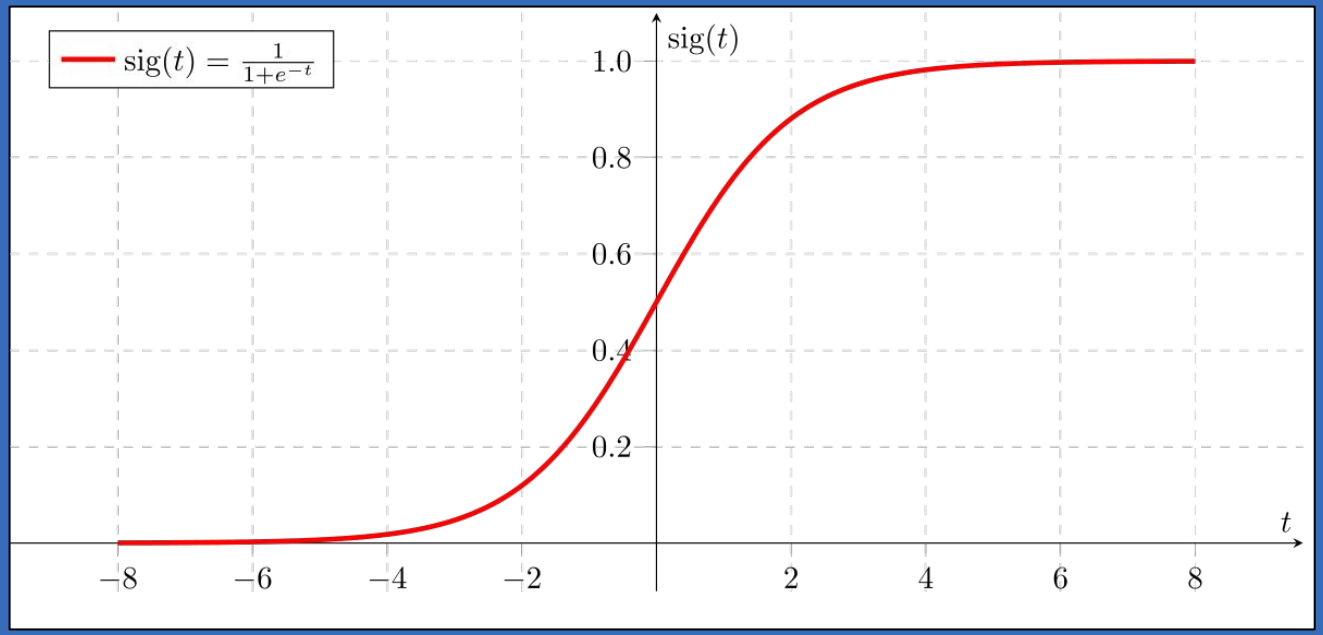
For convenience, you can use a 7-oz (200 mL) can of sliced mushrooms, drained, instead of the fresh mushrooms. Add them after the prosciutto has cooked for 5 to 7 minutes and is almost crisp.

1. Using a spiralizer, cut zucchini into thin strands. Set aside.
2. In a large skillet, heat oil over medium heat. Add mushrooms and prosciutto; cook, stirring, for 6 to 8 minutes or until mushrooms are softened and prosciutto is slightly crisp. Set aside.
3. In a medium saucepan, bring cream and butter to a gentle boil over medium-high heat. Reduce heat and simmer, stirring, for 3 to 5 minutes or until sauce begins to thicken. Whisk in ½ cup (75 mL) cheese and simmer, stirring, for 1 to 2 minutes or until creamy and smooth. Remove from heat and stir in 2 tbsp (30 mL) cheese, salt (if using), pepper and nutmeg.
4. Add zucchini, peas and cream sauce to the skillet, tossing gently. Return skillet to low heat and simmer, stirring, for 2 to 3 minutes or until zucchini is cooked to desired tenderness. Serve garnished with the remaining cheese.





Course Logistics



Lectures

- Monday → Thursday, **1:00 PM- 2:15 PM** in Ed Sci 228 (this room)
- **8 weeks** (August 8th is the last day of class)

Image source: https://thenationonlineng.net/possessed-student-disrupts-lecture/

Attend class

- Small percentage of your grade, but…
- **No way to get the material you missed**
  - Most of class is code examples…
- If something is preventing you from attending, **let me know**
  - Illness, etc → email me
  - Lectures are boring/too slow/too fast/not useful → anonymous feedback form

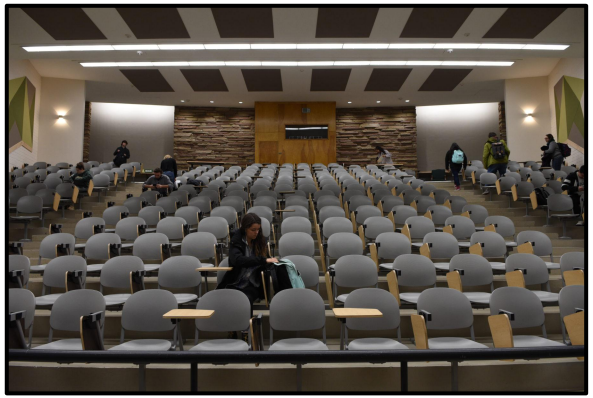


Image source: https://collegian.com/2019/02/ziel-skipping-class-may-have-serious-consequences/

No electronics in class

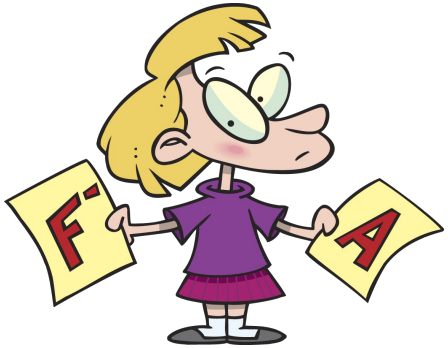
Image source: https://www.youtube.com/watch?v=rQBRpVuQSzk

Programming Assignments

- **2** assignments every week
  - Out **Monday** after class, due **Thursday** by 11:59 PM
  - Out **Thursday** after class, due **Monday** by 11:59 PM
- Turn in on Canvas
- (**Po is out now**, due Thursday at 11:59 PM…)

Assessment

- **Programming Assignments:** (90%)
- **Midterm Exam:** (4%)
  - In class on **July 11th**
- **Final Exam:** (5%)
  - In class on **August 8th**
- **Participation:** (1*%)
  - Show up to class, ask questions, come to office hours, etc

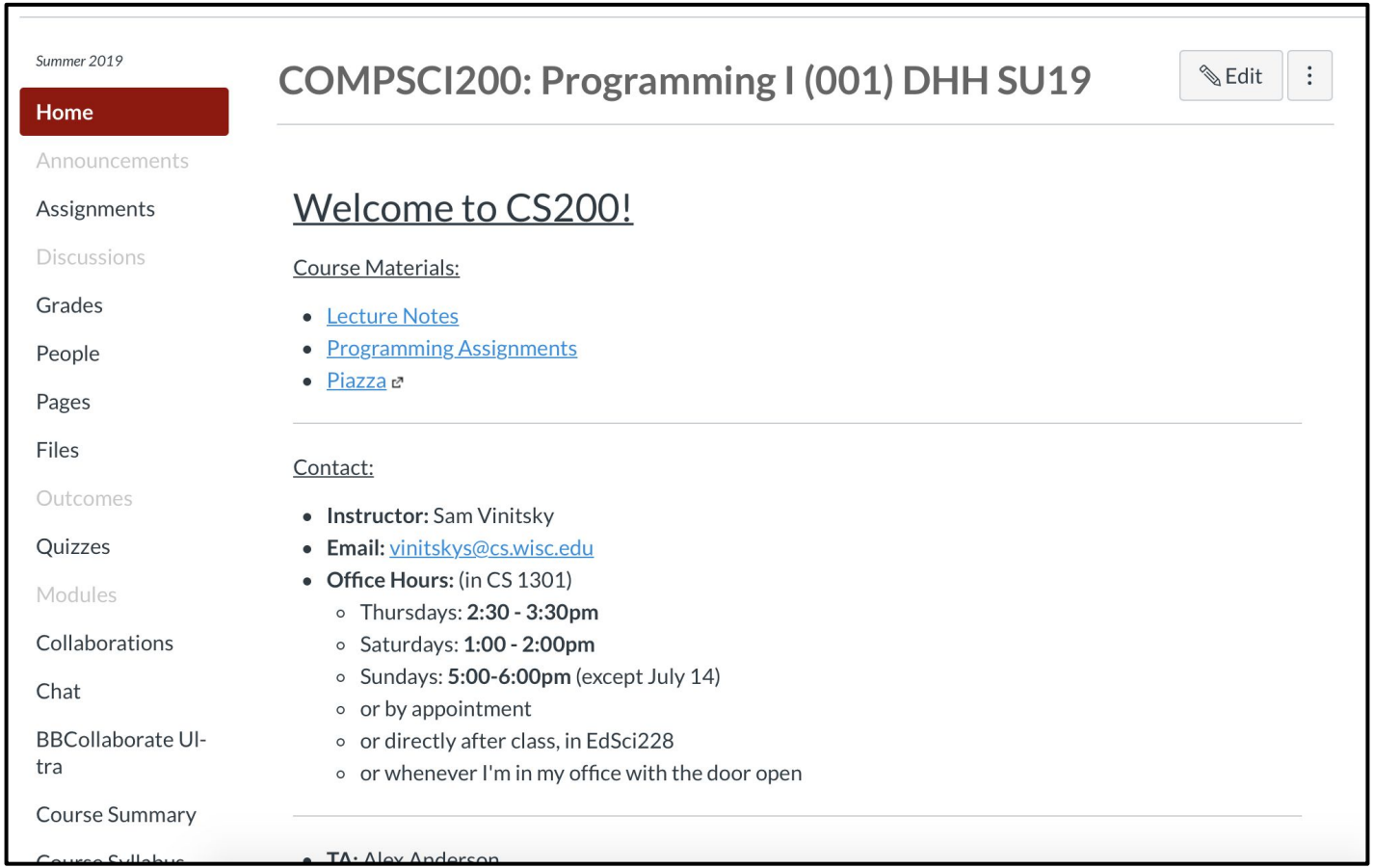


Turn in assignments on time

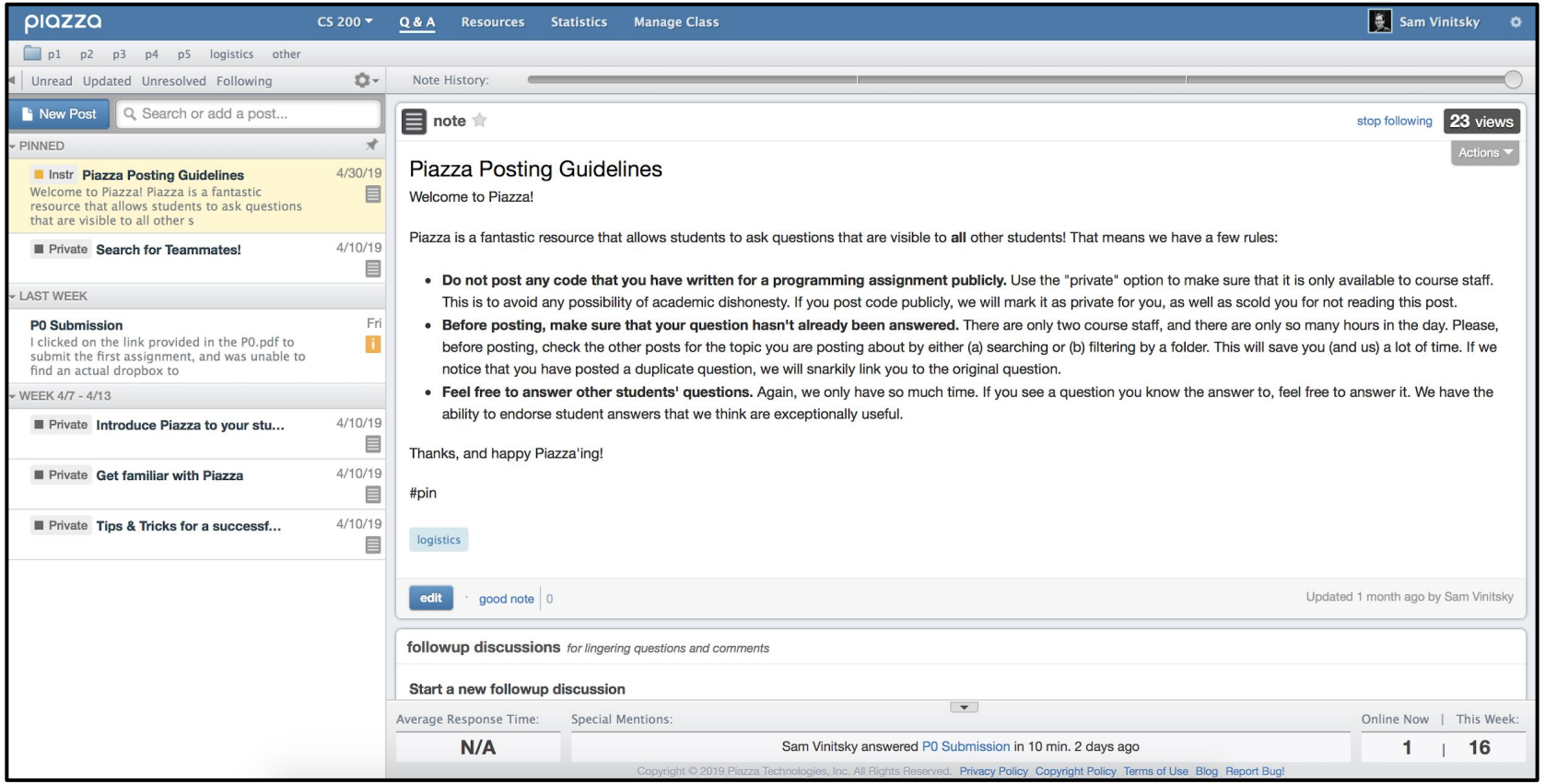
- **No late assignments will be accepted**
- **Slip days:**
  - Each “slip day” extends the due date by 24 hours
  - 5 total
  - 2 on any given assignment
- Slip day form on Canvas

Image source: https://www.walmart.com/ip/Learning-Resources-Big-Time-Student-Clock-12-Hour/24886385

Canvas


Summer 2019
Home
Announcements
Assignments
Discussions
Grades
People
Pages
Files
Outcomes
Quizzes
Modules
Collaborations
Chat
BBCollaborate Ul-tra
Course Summary
COMPSCI200: Programming I (001) DHH SU19
Edit
Welcome to CS200!
Course Materials:
Lecture Notes
Programming Assignments
Piazza
Contact:
Instructor: Sam Vinitsky
Email: vinitskys@cs.wisc.edu
Office Hours: (in CS 1301)
Thursdays: 2:30 - 3:30pm
Saturdays: 1:00 - 2:00pm
Sundays: 5:00-6:00pm (except July 14)
or by appointment
or directly after class, in EdSci228
or whenever I'm in my office with the door open

Piazza


piazza
CS 200
Q & A
Resources
Statistics
Manage Class
Sam Vinitsky
p1 p2 p3 p4 p5 logistics other
Unread Updated Unresolved Following
Note History:
New Post
Search or add a post...
PINNED
Instr Piazza Posting Guidelines
4/30/19
Welcome to Piazza! Piazza is a fantastic resource that allows students to ask questions that are visible to all other s
Private Search for Teammates!
4/10/19
LAST WEEK
P0 Submission
Fri
I clicked on the link provided in the P0.pdf to submit the first assignment, and was unable to find an actual dropbox to
WEEK 4/7 - 4/13
Private Introduce Piazza to your stu...
4/10/19
Private Get familiar with Piazza
4/10/19
Private Tips & Tricks for a successf...
4/10/19
note
stop following
23 views
Actions
Piazza Posting Guidelines
Welcome to Piazza!
Piazza is a fantastic resource that allows students to ask questions that are visible to all other students! That means we have a few rules:
Do not post any code that you have written for a programming assignment publicly. Use the "private" option to make sure that it is only available to course staff. This is to avoid any possibility of academic dishonesty. If you post code publicly, we will mark it as private for you, as well as scold you for not reading this post.
Before posting, make sure that your question hasn't already been answered. There are only two course staff, and there are only so many hours in the day. Please, before posting, check the other posts for the topic you are posting about by either (a) searching or (b) filtering by a folder. This will save you (and us) a lot of time. If we notice that you have posted a duplicate question, we will snarkily link you to the original question.
Feel free to answer other students' questions. Again, we only have so much time. If you see a question you know the answer to, feel free to answer it. We have the ability to endorse student answers that we think are exceptionally useful.
Thanks, and happy Piazza'ing!
#pin
logistics
edit
good note 0
Updated 1 month ago by Sam Vinitsky
followup discussions for lingering questions and comments
Start a new followup discussion
Average Response Time:
N/A
Special Mentions:
Sam Vinitsky answered P0 Submission in 10 min. 2 days ago
Online Now | This Week:
1 | 16
Copyright © 2019 Piazza Technologies, Inc. All Rights Reserved. Privacy Policy Copyright Policy Terms of Use Blog Report Bug!

Office hours

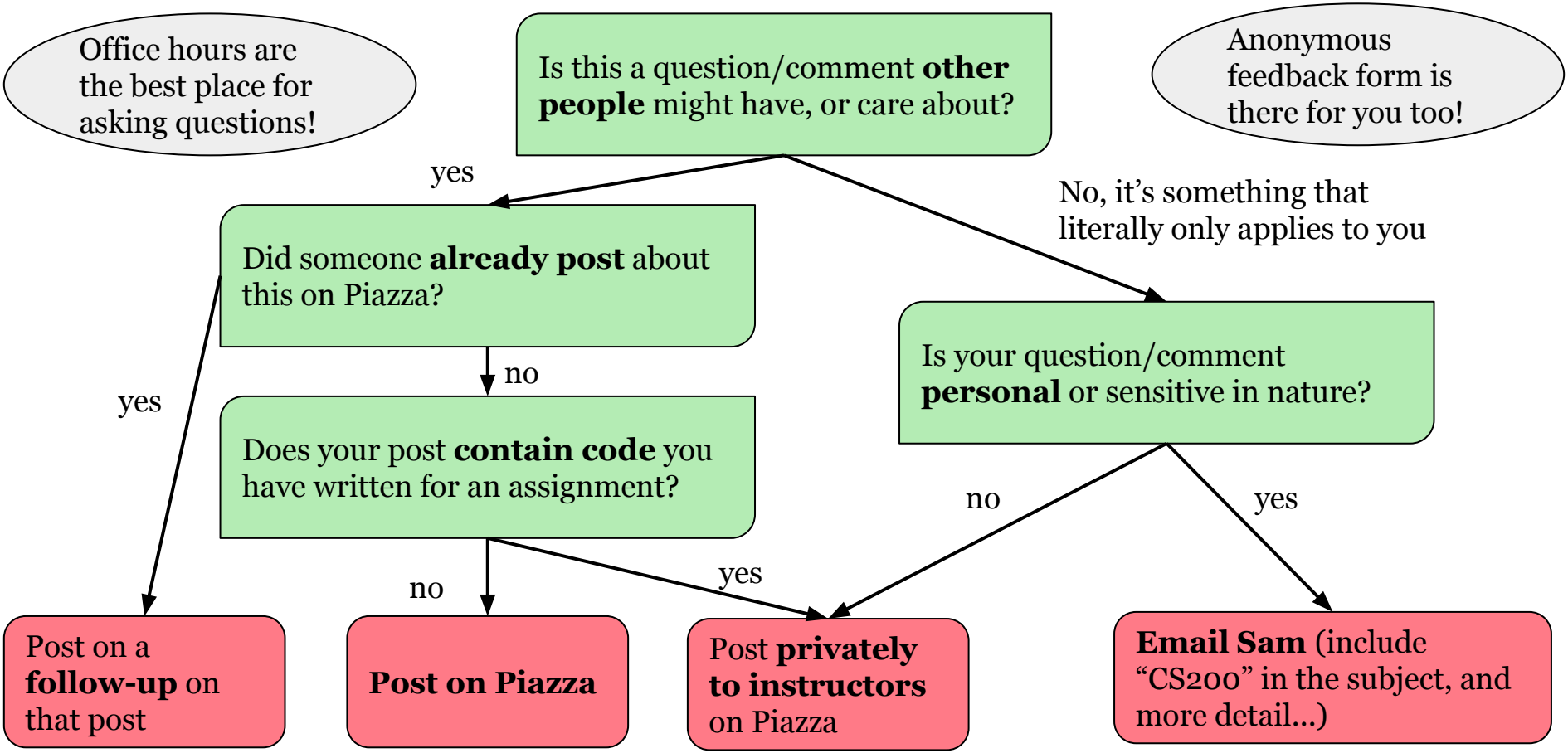
Instructor: Sam Vinitsky
Email: vinitskys@cs.wisc.edu
Office Hours: (in CS 1301)

- Thursdays: **2:30 - 3:30 pm**
- Saturdays: **1:00 - 2:00 pm**
- Sundays: **5:00 - 6:00 pm** [except July 14]
- or by appointment
- or directly after class, in EdSci228
- or whenever I'm in my office with the door open

TA: Alex Anderson (aanderson65@wisc.edu)
Email: aanderson65@wisc.edu
Office Hours: (in CS 1304)

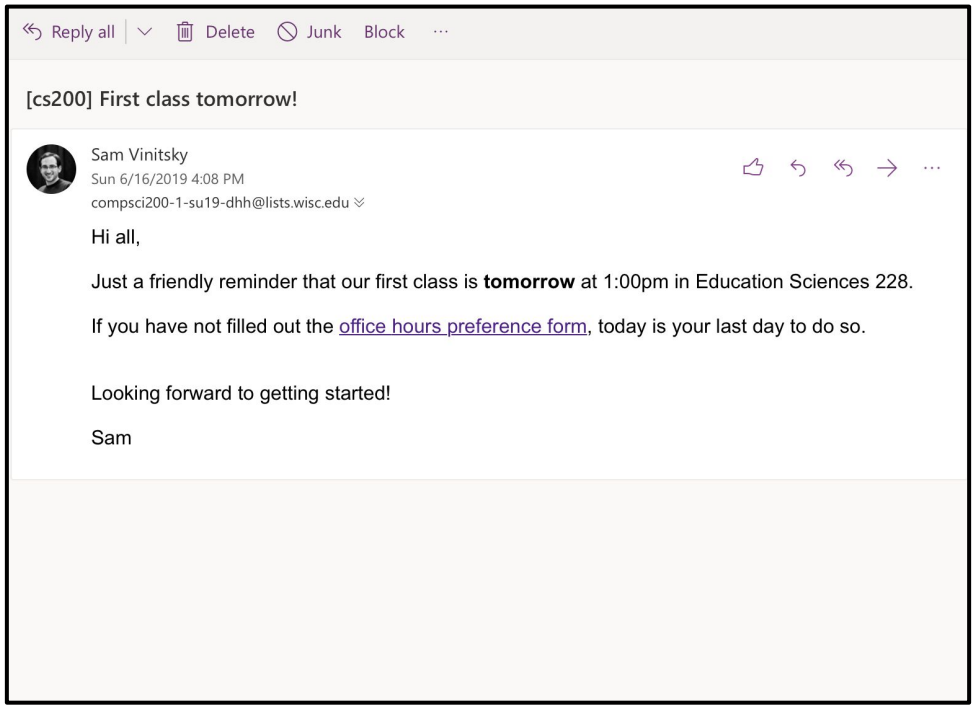
- Mondays: **2:30 - 3:30 pm**
- Tuesdays: **11:30 am - 12:30 pm**
- Wednesday: **2:30 - 3:30 pm**
- Thursday: **10:30 - 11:30 am**
- or by appointment

Asking questions online:

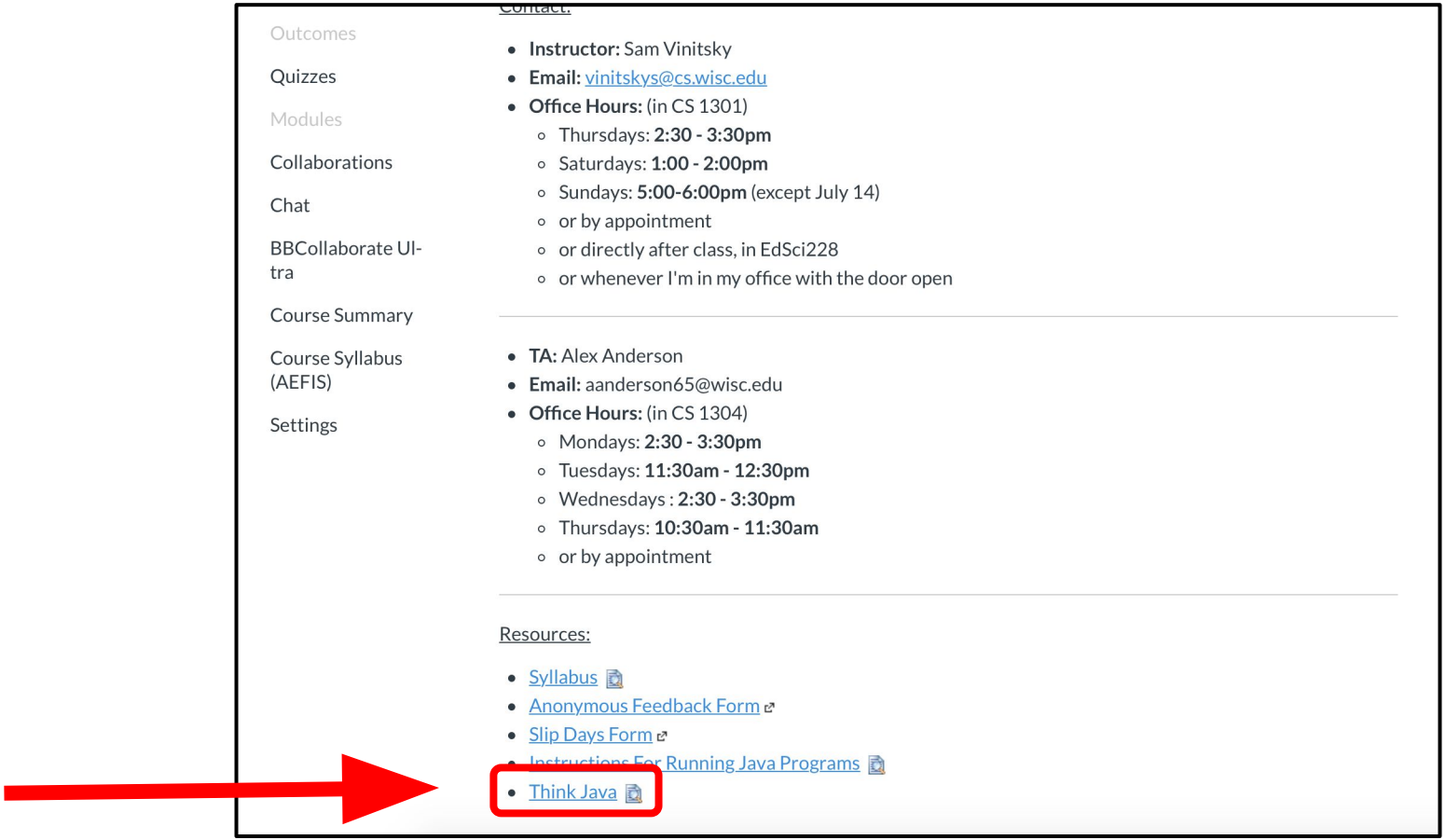


How will Sam communicate with me?

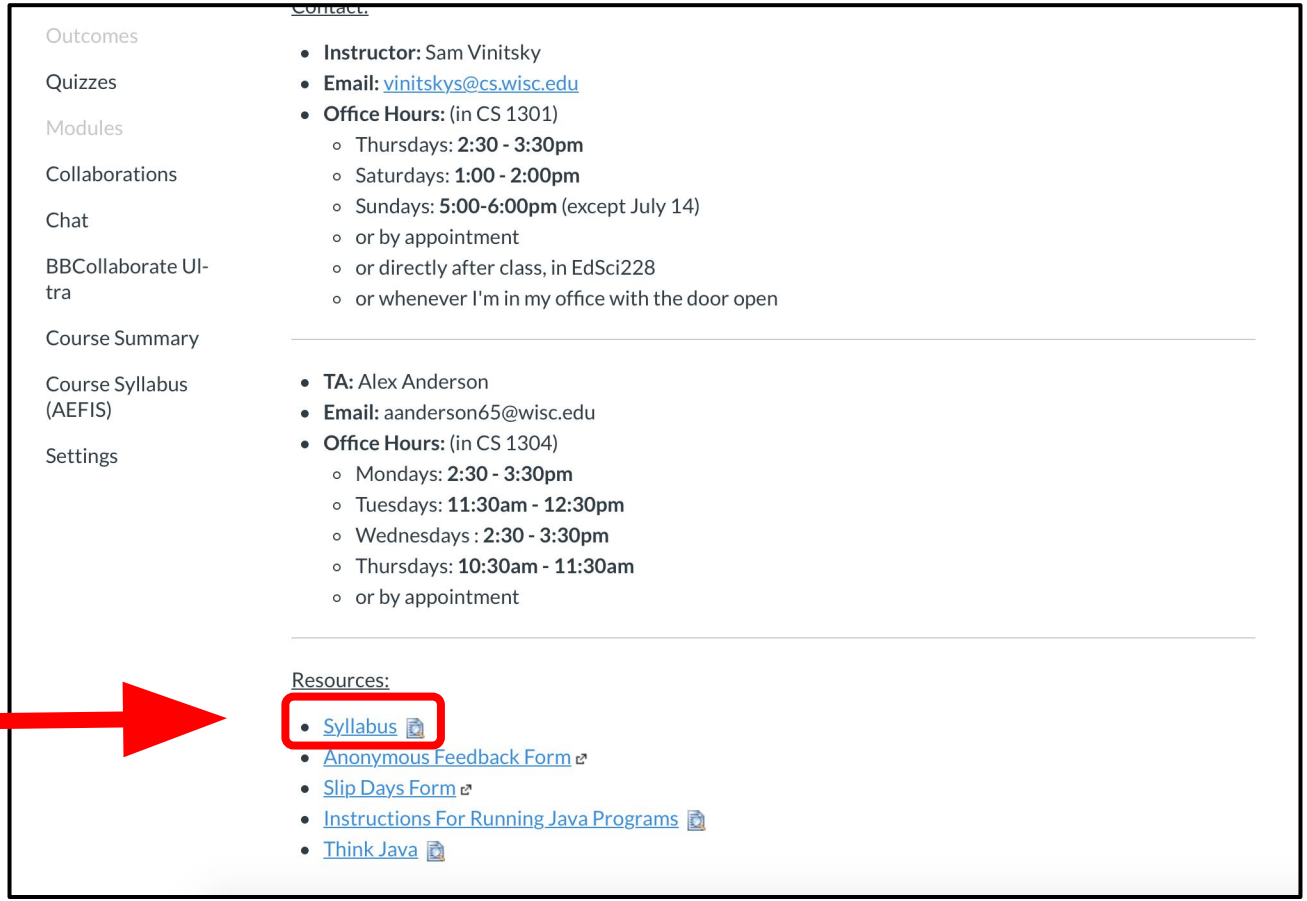
- **Email list!**
  - If you did not get my email yesterday, tell me ASAP (via email)



"Think Java" -- free textbook!



Read the syllabus…



You are welcome here.

Tips for success in cs200

Difficulty warning!

- This class will be **hard** and **a lot of work**
  - This class is usually 15 weeks…
  - Learning a new **way of thinking**
  - Learning a new “language”
  - Computers…

Image source: https://www.wabisabilearning.com/blog/future-learning-social-networking-beyond

Tip #1: Start early

- Start assignments the **day** the are assigned
- Assignments will be **hard**, and will take a long time (worth 90%...)
  - It's impossible* to estimate the amount of time a program will take
- We're here to help, but not all the time...

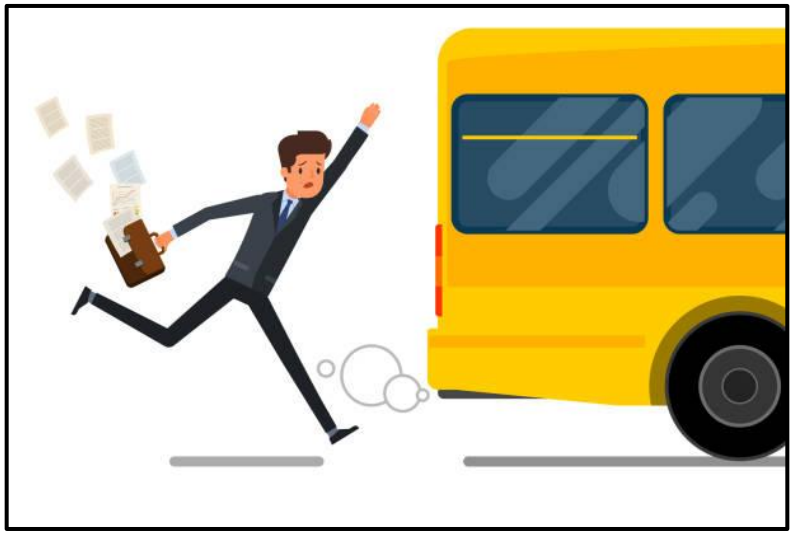


Image source: https://www.123rf.com/photo_78618604_stock-vector-concept-of-lateness-.html

Tip #2: Ask questions

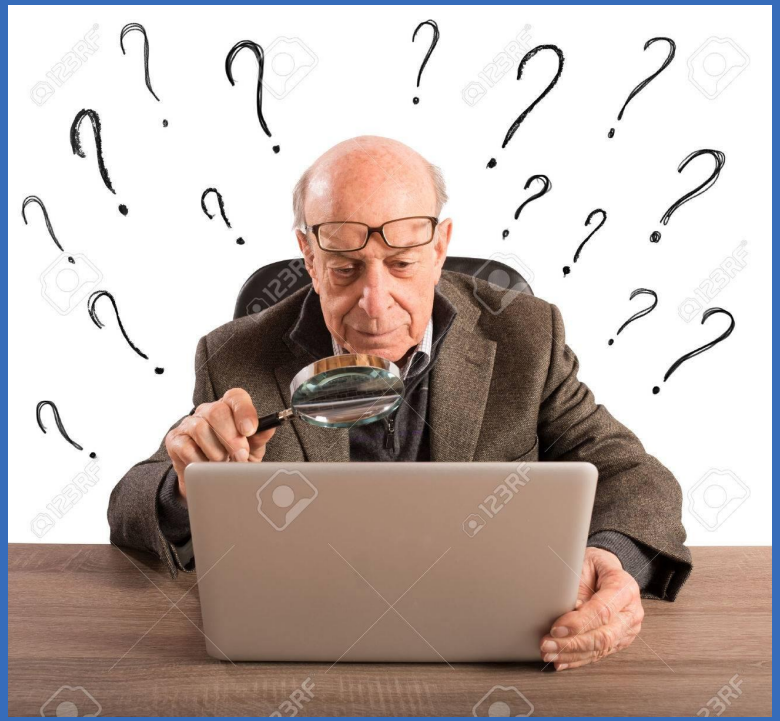
- Ask questions in **lecture**
- Ask questions about the **assignments**
  - Piazza!
- Ask questions of **Java**
  - Just try things…
  - Experimentation is **key**

Image source: https://www.macworld.com/article/3077986/your-top-questions-to-mac-911-and-some-answers.html

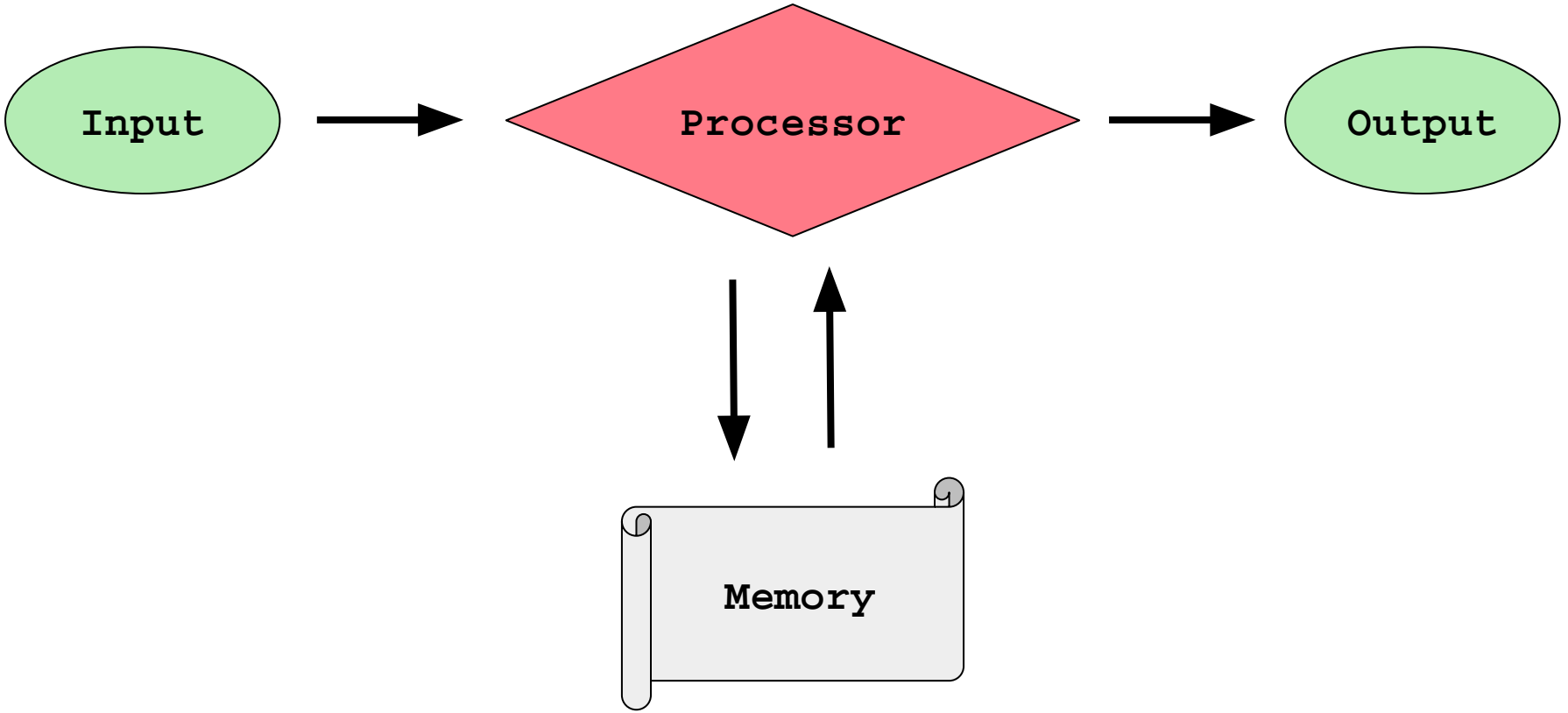
Tip #3: Have fun!

www.shutterstock.com • 1056496400

How do computers work?



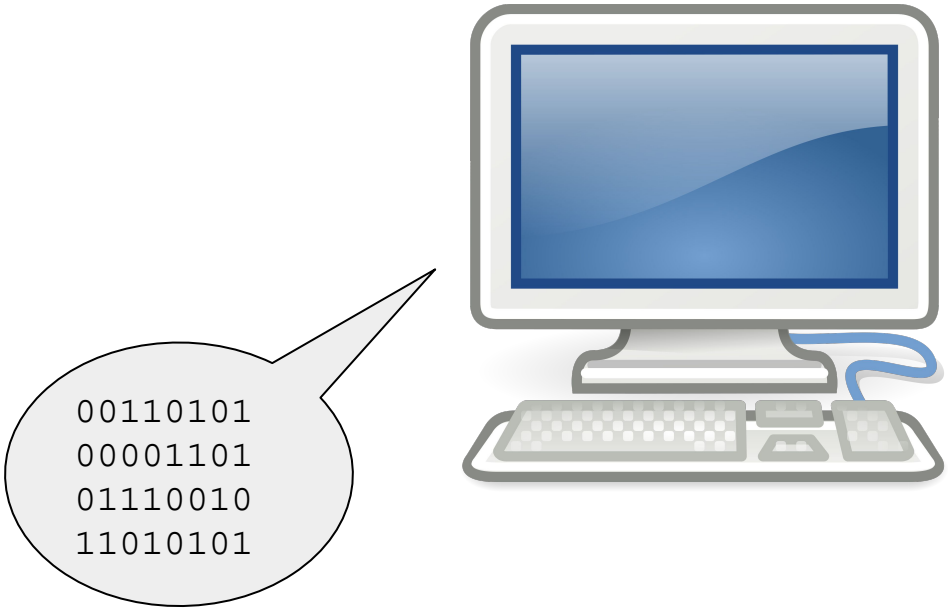
How do computers work?



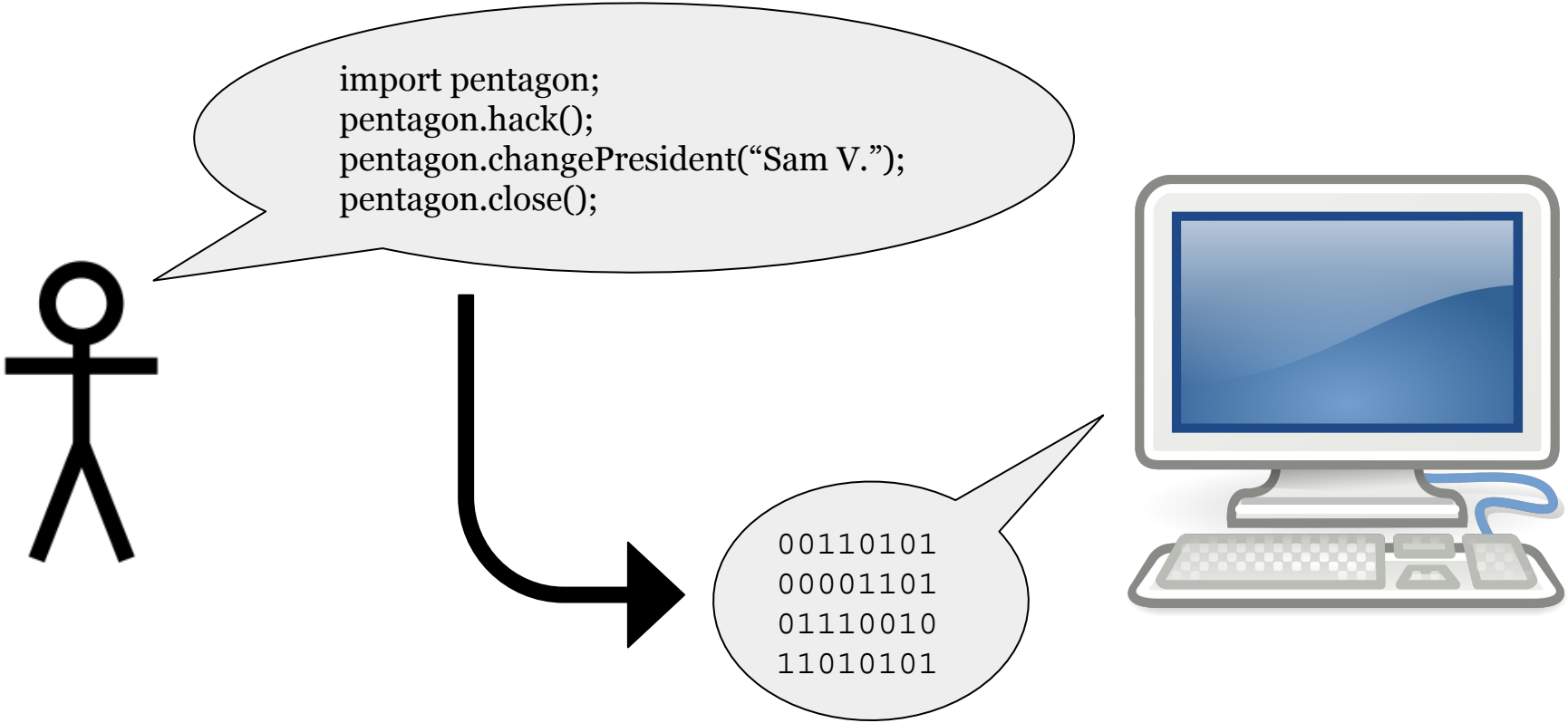
What does memory look like?

```
...
75:          01101101
76:          11010101
77:          01011101
78:          11110010
79:          01010001
80:          01110110
81:          01110111
82:          11010110
83:          11010110
84:          01110110
85:          00000000
86:          00000000
87:          00000000
88:          00000000
89:          01110101
...
```

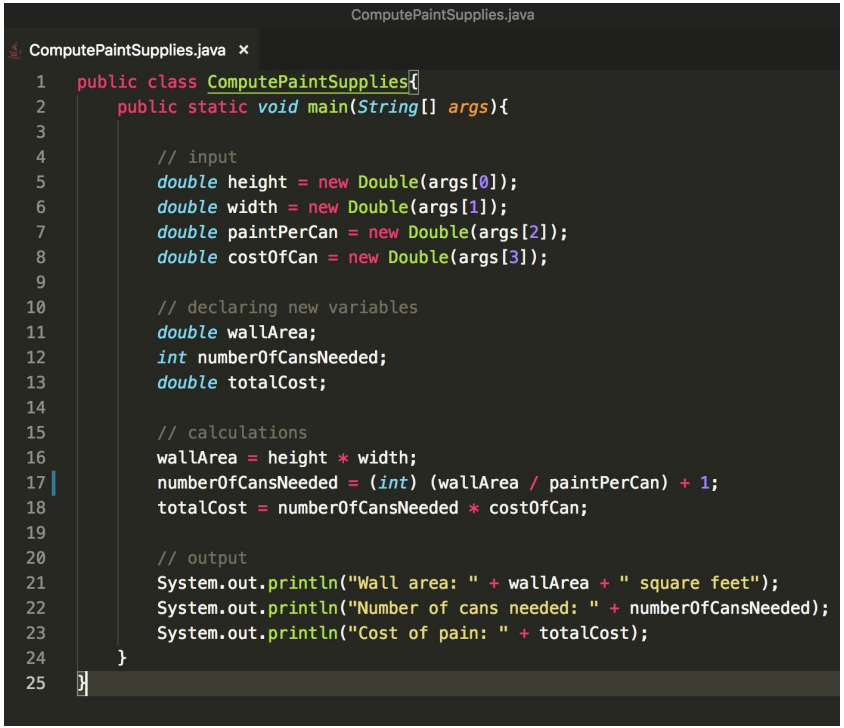
Problem: I don't speak binary



Solution: Programming languages



Java programming flow: Write



ComputePaintSupplies.java

Java programming flow: Compile

ComputePaintSupplies.java

```java
public class ComputePaintSupplies{
    public static void main(String[] args){

        // input
        double height = new Double(args[0]);
        double width = new Double(args[1]);
        double paintPerCan = new Double(args[2]);
        double costOfCan = new Double(args[3]);

        // declaring new variables
        double wallArea;
        int numberOfCansNeeded;
        double totalCost;

        // calculations
        wallArea = height * width;
        numberOfCansNeeded = (int) (wallArea / paintPerCan) + 1;
        totalCost = numberOfCansNeeded * costOfCan;

        // output
        System.out.println("Wall area: " + wallArea + " square feet");
        System.out.println("Number of cans needed: " + numberOfCansNeeded);
        System.out.println("Cost of pain: " + totalCost);
    }
}
```

compile

```
cafe babe 0000 0034 003c 0a00 1200 1b07
001c 0a00 0200 1d0a 0002 001e 0900 1f00
2007 0021 0a00 0600 1b08 0022 0a00 0600
230a 0006 0024 0800 250a 0006 0026 0a00
2700 2808 0029 0a00 0600 2a08 002b 0700
2c07 002d 0100 063c 696e 6974 3e01 0003
2829 5601 0004 436f 6465 0100 0f4c 696e
654e 756d 6265 7254 6162 6c65 0100 046d
6169 6e01 0016 285b 4c6a 6176 612f 6c61
6e67 2f53 7472 696e 673b 2956 0100 0a53
6f75 7263 6546 696c 6501 0019 436f 6d70
7574 6550 6169 6e74 5375 7070 6c69 6573
2e6a 6176 610c 0013 0014 0100 106a 6176
612f 6c61 6e67 2f44 6f75 626c 650c 0013
002e 0c00 2f00 3007 0031 0c00 3200 3301
0017 6a61 7661 2f6c 616e 672f 5374 7269
6e67 4275 696c 6465 7201 000b 5761 6c6c
2061 7265 613a 200c 0034 0035 0c00 3400
3601 000c 2073 7175 6172 6520 6665 6574
0c00 3700 3807 0039 0c00 3a00 2e01 0017
4e75 6d62 6572 206f 6620 6361 6e73 206e
6565 6465 643a 200c 0034 003b 0100 0e43
6f73 7420 6f66 2070 6169 6e3a 2001 0014
436f 6d70 7574 6550 6169 6e74 5375 7070
6c69 6573 0100 106a 6176 612f 6c61 6e67
2f4f 626a 6563 7401 0015 284c 6a61 7661
2f6c 616e 672f 5374 7269 6e67 3b29 5601
000b 646f 7562 6c65 5661 6c75 6501 0003
2829 4401 0010 6a61 7661 2f6c 616e 672f
5379 7374 656d 0100 036f 7574 0100 154c
6a61 7661 2f69 6f2f 5072 696e 7453 7472
6561 6d3b 0100 0661 7070 656e 6401 002d
284c 6a61 7661 2f6c 616e 672f 5374 7269
6e67 3b29 4c6a 6176 612f 6c61 6e67 2f53
7472 696e 6742 7569 6c64 6572 3b01 001c
2844 294c 6a61 7661 2f6c 616e 672f 5374
7269 6e67 4275 696c 6465 723b 0100 0874
6f53 7472 696e 6701 0014 2829 4c6a 6176
```

ComputePaintSupplies.java

ComputePaintSupplies.class

Java programming flow: Run

ComputePaintSupplies.java

```java
public class ComputePaintSupplies{
    public static void main(String[] args){

        // input
        double height = new Double(args[0]);
        double width = new Double(args[1]);
        double paintPerCan = new Double(args[2]);
        double costOfCan = new Double(args[3]);

        // declaring new variables
        double wallArea;
        int numberOfCansNeeded;
        double totalCost;

        // calculations
        wallArea = height * width;
        numberOfCansNeeded = (int) (wallArea / paintPerCan) + 1;
        totalCost = numberOfCansNeeded * costOfCan;

        // output
        System.out.println("Wall area: " + wallArea + " square feet");
        System.out.println("Number of cans needed: " + numberOfCansNeeded);
        System.out.println("Cost of pain: " + totalCost);
    }
}
```

ComputePaintSupplies.java

compile

Executed by "Java virtual machine"

```
cafe babe 0000 0034 003c 0a00 1200 1b07
001c 0a00 0200 1d0a 0002 001e 0900 1f00
2007 0021 0a00 0600 1b08 0022 0a00 0600
230a 0006 0024 0800 250a 0006 0026 0a00
2700 2808 0029 0a00 0600 2a08 002b 0700
2c07 002d 0100 063c 696e 6974 3e01 0003
2829 5601 0004 436f 6465 0100 0f4c 696e
654e 756d 6265 7254 6162 6c65 0100 046d
6169 6e01 0016 285b 4c6a 6176 612f 6c61
6e67 2f53 7472 696e 673b 2956 0100 0a53
6f75 7263 6546 696c 6501 0019 436f 6d70
7574 6550 6169 6e74 5375 7070 6c69 6573
2e6a 6176 610c 0013 0014 0100 106a 6176
612f 6c61 6e67 2f44 6f75 626c 650c 0013
002e 0c00 2f00 3007 0031 0c00 3200 3301
0017 6a61 7661 2f6c 616e 672f 5374 7269
6e67 4275 696c 6465 7201 000b 5761 6c6c
2061 7265 613a 200c 0034 0035 0c00 3400
3601 000c 2073 7175 6172 6520 6665 6574
0c00 3700 3807 0039 0c00 3a00 2e01 0017
4e75 6d62 6572 206f 6620 6361 6e73 206e
6565 6465 643a 200c 0034 003b 0100 0e43
6f73 7420 6f66 2070 6169 6e3a 2001 0014
436f 6d70 7574 6550 6169 6e74 5375 7070
6c69 6573 0100 106a 6176 612f 6c61 6e67
2f4f 626a 6563 7401 0015 284c 6a61 7661
2f6c 616e 672f 5374 7269 6e67 3b29 5601
000b 646f 7562 6c65 5661 6c75 6501 0003
2829 4401 0010 6a61 7661 2f6c 616e 672f
5379 7374 656d 0100 036f 7574 0100 154c
6a61 7661 2f69 6f2f 5072 696e 7453 7472
6561 6d3b 0100 0661 7070 656e 6401 002d
284c 6a61 7661 2f6c 616e 672f 5374 7269
6e67 3b29 4c6a 6176 612f 6c61 6e67 2f53
7472 696e 6742 7569 6c64 6572 3b01 001c
2844 294c 6a61 7661 2f6c 616e 672f 5374
7269 6e67 4275 696c 6465 723b 0100 0874
6f53 7472 696e 6701 0014 2829 4c6a 6176
```

ComputePaintSupplies.class

Java programming flow: Summary

1. **Write** your program (using a text editor) in Java
2. **Compile** your program into bytecode (`.class` file)
3. **Run** your code!

- How to do this in Visual Studio Code?

Java program structure

```java
public class Welcome {

    public static void main(String[] args){
        System.out.println("Welcome to CS200!");
        System.out.println("Good luck!");
        // you're gonna need it...
    }

}
```

Java program structure

- **“Class declaration”**
- Every file starts with this
- `public class` → always
- Green word = “Class name”
  - Same as file name
  - (`Welcome.java`)
- Everything must be inside the curly braces

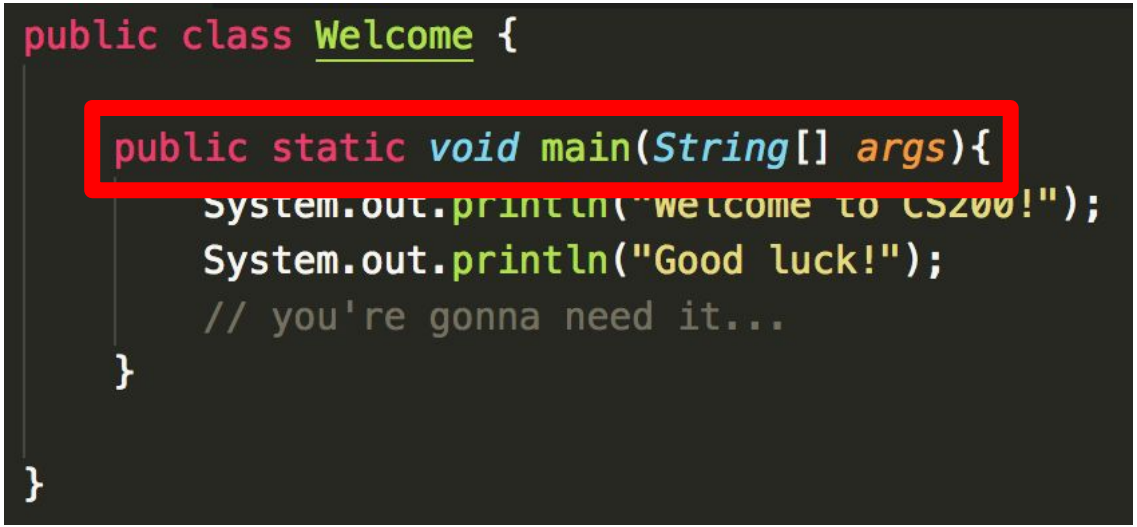
```
public class Welcome {

    public static void main(String[] args){
        System.out.println("Welcome to CS200!");
        System.out.println("Good luck!");
        // you're gonna need it...
    }

}
```

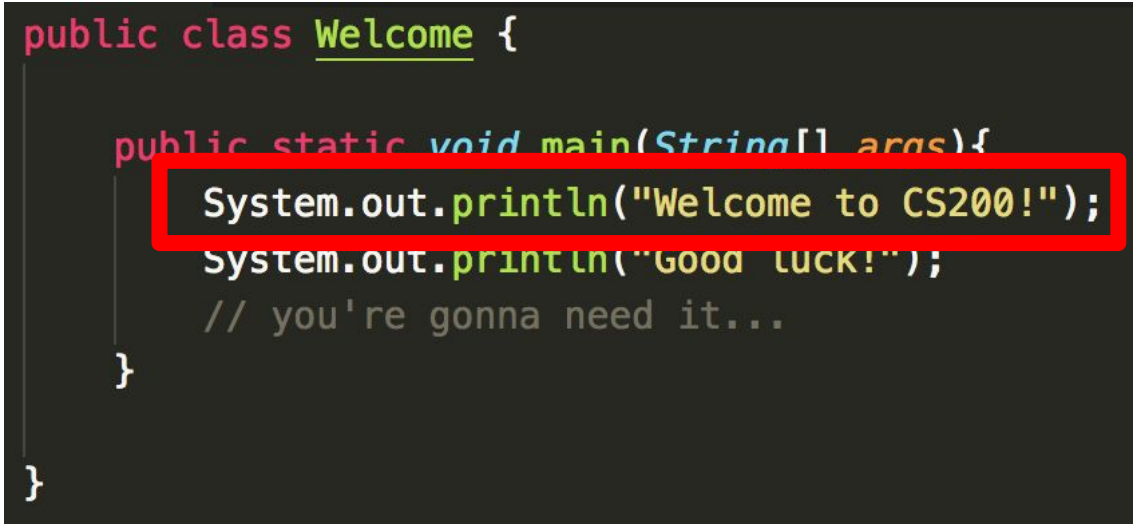
Java program structure

- **“`main` function”**
- Needs to look EXACTLY like this.
- We’ll explain more on Thursday.
- Pair of curly braces
- When the program is run, the code written in “`main`” will get executed line by line



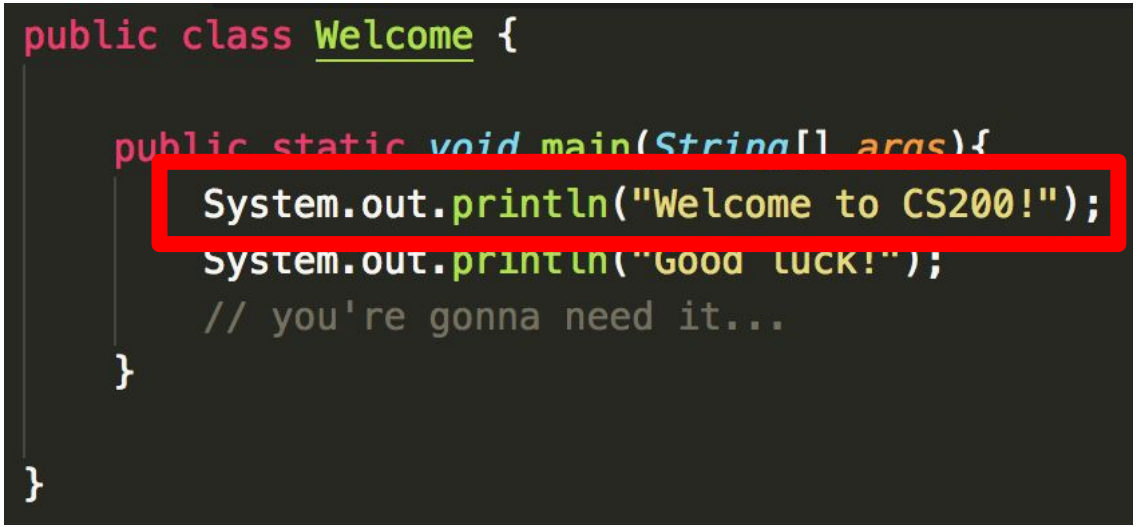
Java program structure

- **“statement”**
- Line of code to be executed
- Ends in a semicolon “;”
- This one will output “**Welcome to CS200**” to the screen
- Happens first



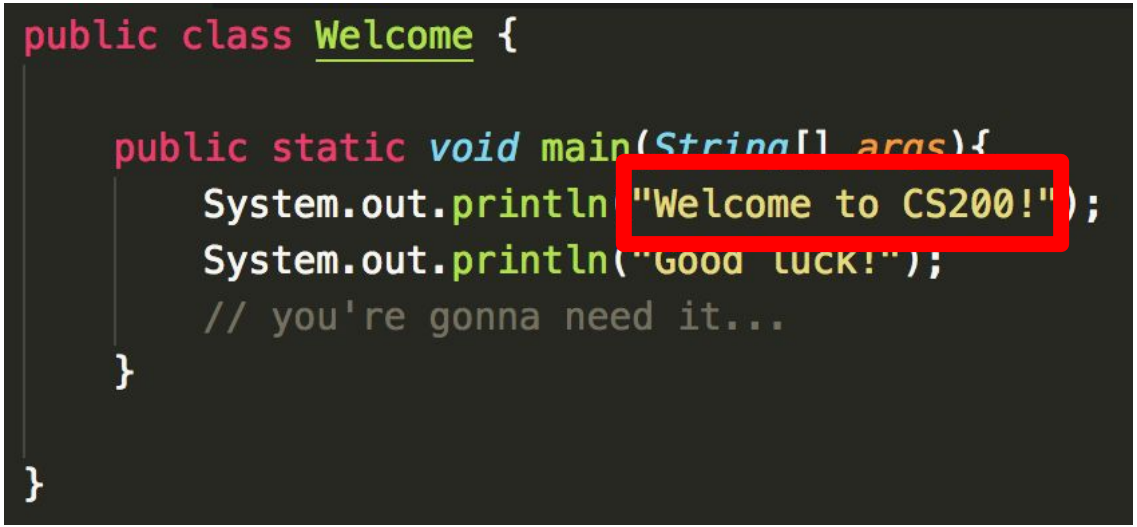
Java program structure

- **“print statement”**
- Outputs the stuff between the parentheses



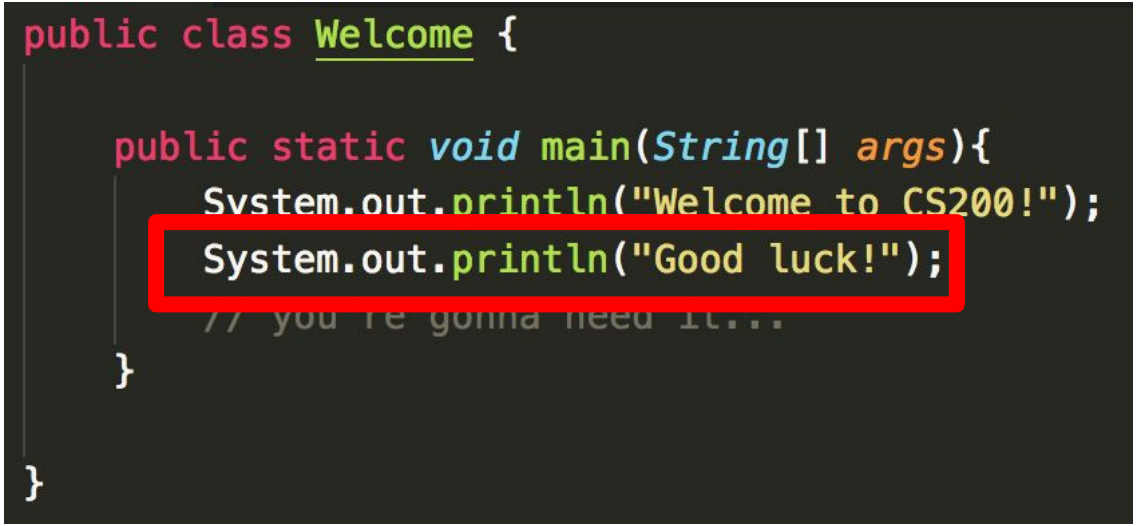
Java program structure

- **“string”**
- Quotes tell Java that these are words, not code



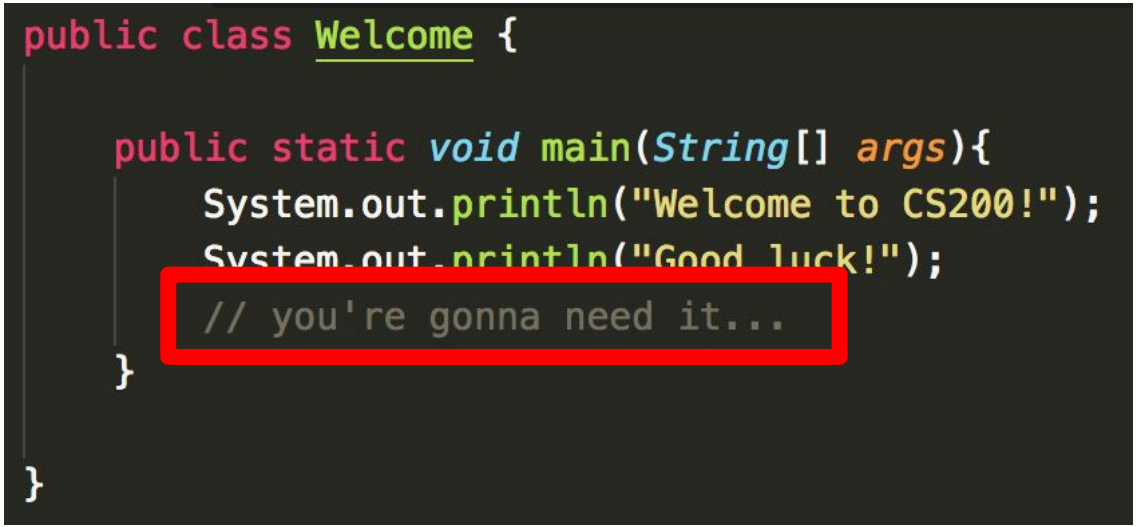
Java program structure

- Another **“statement”**
- This one will output “**Good luck!**” to the screen
- Happens second



Java program structure

- **“Comment”**
- Ignored
- “Note to myself”



Ex: Goodbye!

- **Task:** Write a program named `Goodbye.java` that outputs the phrase "Goodbye Sam!"

Resources...

- **Course Policy:**
  - Syllabus
- **Computers + Java basics:**
  - *Think Java* -- Chapter 1