CSE 121 – Lesson 9

Miya Natsuhara

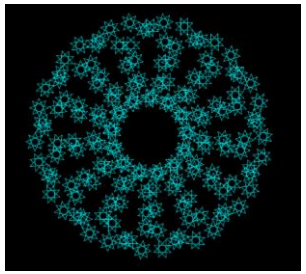
Spring 2023

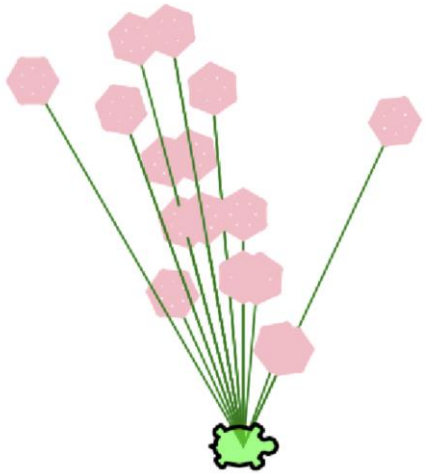
Music: 121 23sp Lecture Vibes

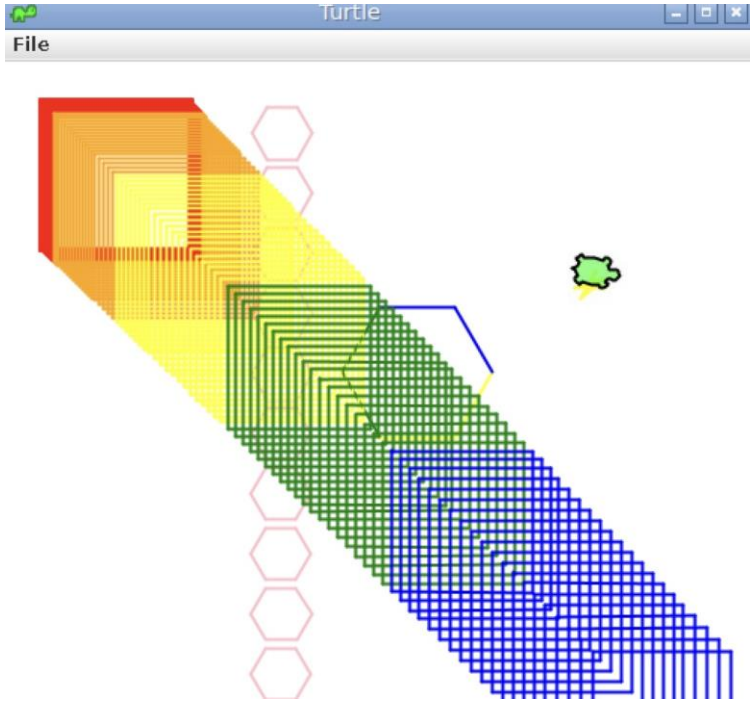
TAs:				
	Jasmine	*Atharva*	*Mia*	*Justin*
	Shananda	*Julia*	*Archit*	*Aishah*
	Vidhi	*Anju*	*Grace*	*Claire*
	Larry	*Lydia*	*Kailye*	*Lydia*
	Jacqueline	*Jonus*	*Joshua*	*Kai*
	Afifah	*Hugh*	*James*	

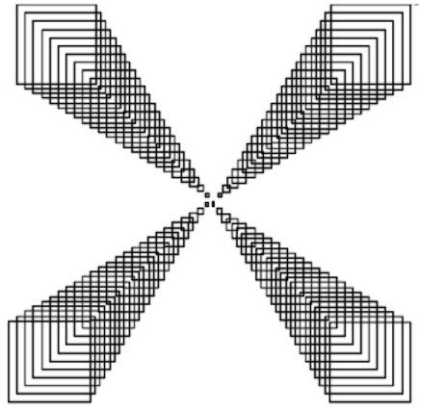
sli.do #cse121

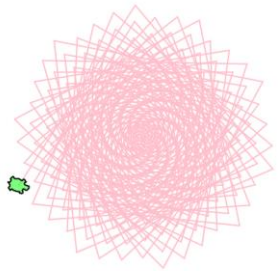
Creative Project Showcase!

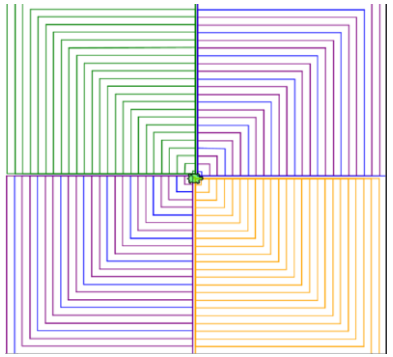


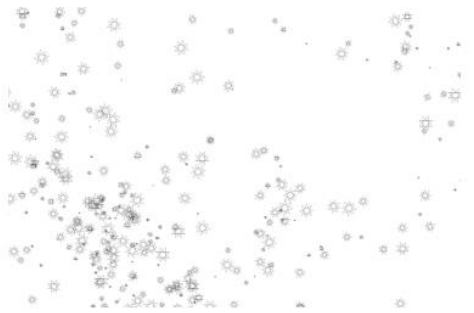












(zoomed-out version)

Announcements, Reminders

- Creative Project 2 released
  - Due Tuesday, May 2
- Resubmission Cycle 2 form released
  - Note: this is the last time C0 is eligible for resubmission.
- Quiz 0 Retakes possible 5/2, 5/9
  - Grades for Quiz 0 Retakes will be released all at once
- Quiz 1 on Thursday, May 4 in quiz section
- Mid-Quarter Formative Feedback with Ken Yasuhara for part of class on Wednesday, May 3

Common Problem-Solving Strategies

- **Analogy** – Is this similar to another problem you've seen?
- **Brainstorming** – Consider steps to solve problem before jumping into code
  - Try to do an example "by hand" → outline steps
- **Solve sub-problems** – Is there a smaller part of the problem to solve?
- **Debugging** – Does your solution behave correctly?
  - What is it doing?
  - What do you expect it to do?
  - What area of your code controls that part of the output?
- **Iterative Development** – Can we start by solving a different problem that is easier?

Metacognition

Metacognition: thinking about how you think

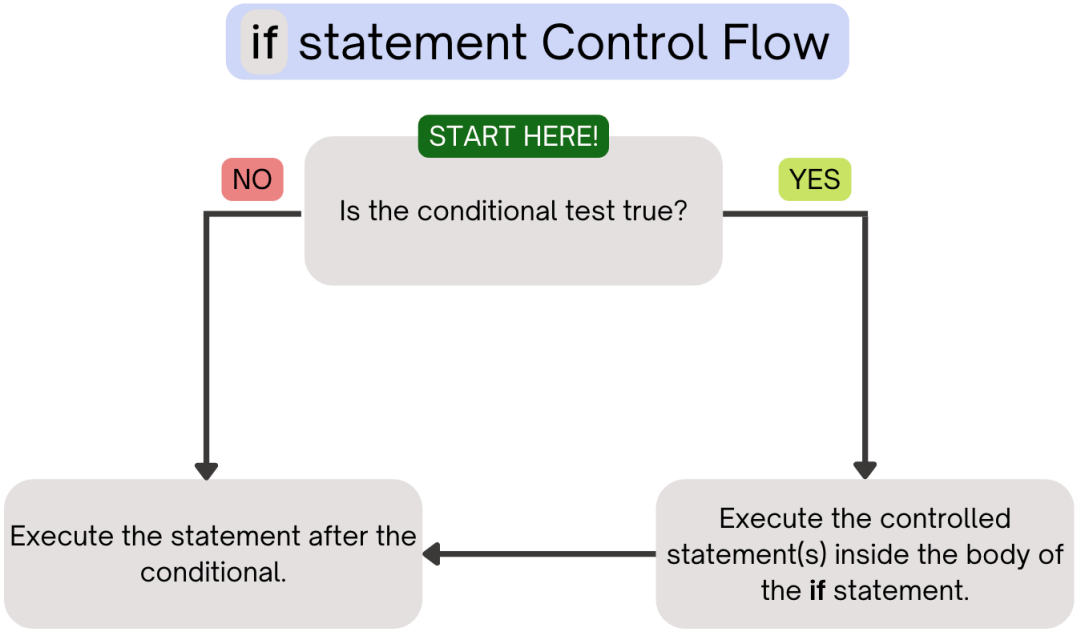
Asking questions about your solution process

Examples

- **While debugging**: explain to yourself why you're making this change to your program
- **Before running your program**: make an explicit prediction of what you expect to see
- **When coding**: be aware of when you're not making progress, so you can take a break or try a different strategy
- **When studying**: What is the relationship of this topic to other ideas in the course?

(PCM) Conditionals

```
if (test) {
    body (statements to be executed)
}
```

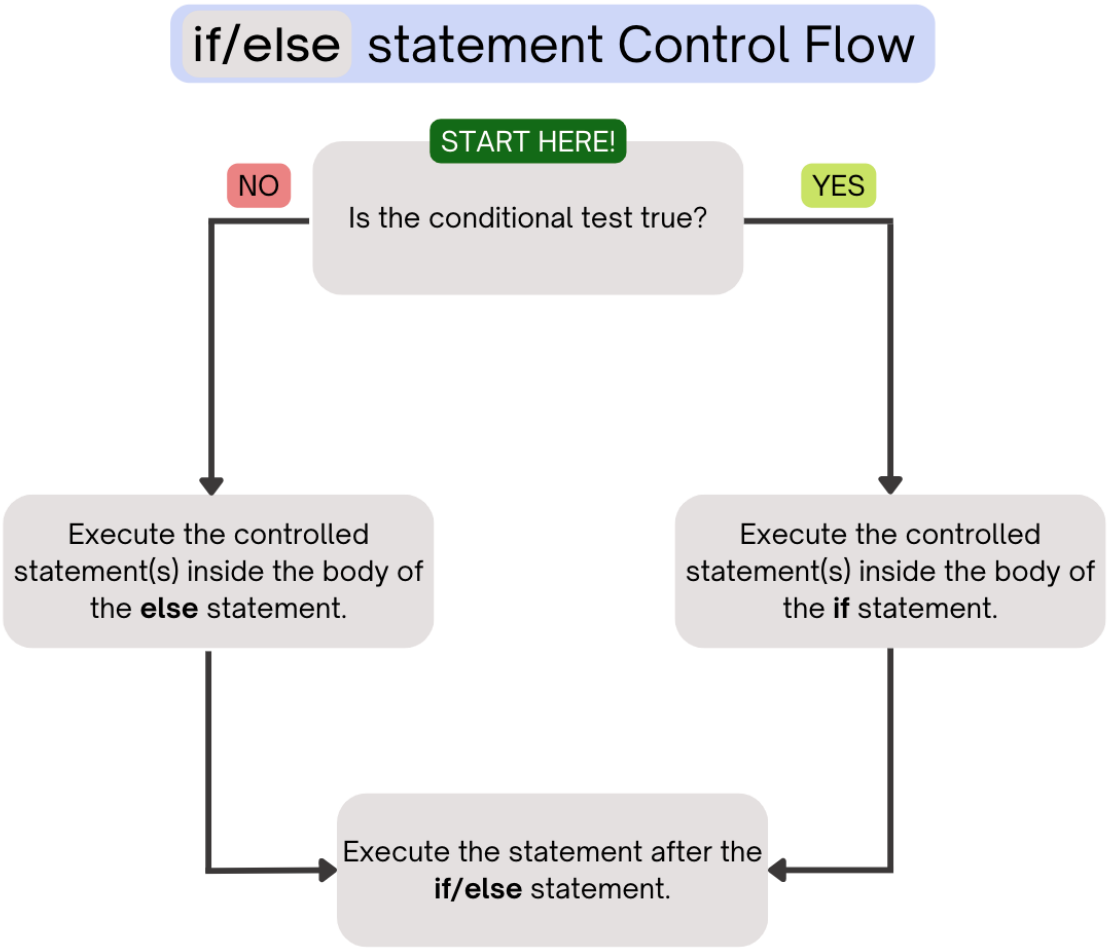


Executes a block of statements only if the test is true

(PCM) Conditionals

```
if ( test ) {
        statement(s)
} else {
        statement(s)
}
```

Executes a block of statements if the test is true, executes another block of statements if the test is false

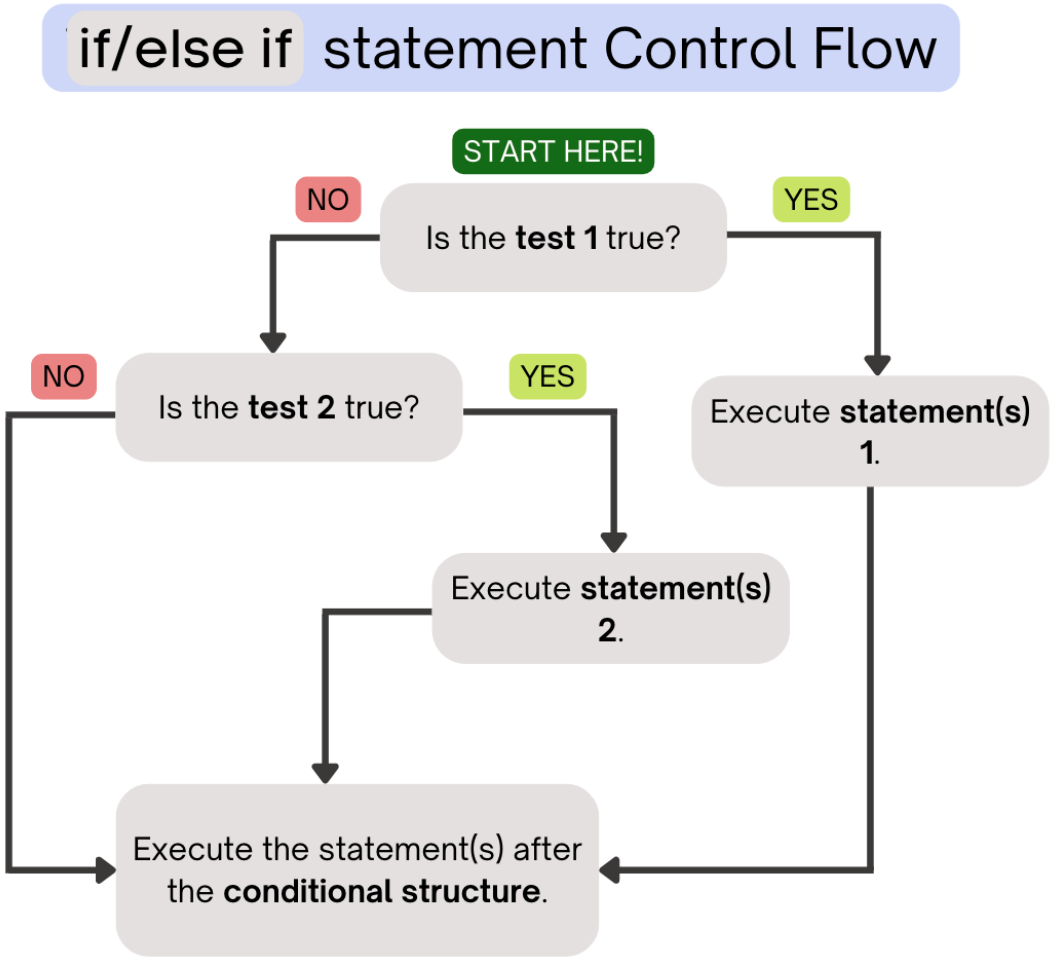


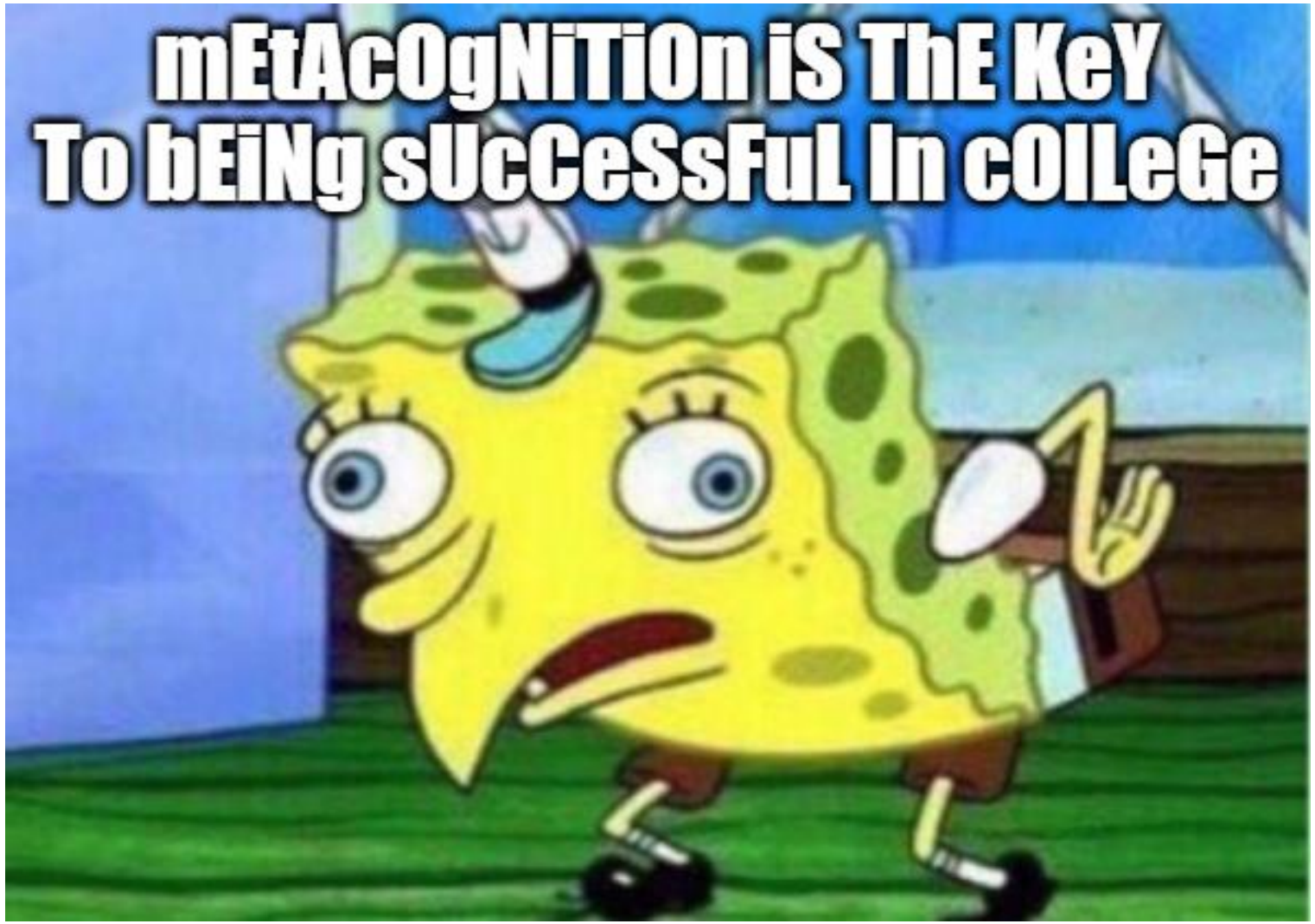
(PCM) Conditionals

```
if ( test ) {
        statement(s)
} else if (test) {
        statement(s)
}
```

Chooses between a block of statements to execute out of multiple choices, depending on which test it passes

- If it ends in an else, exactly one block will be executed.
- If it ends in an else if, at most one block will be executed, but the code also may not execute any blocks of statements.




mEtAcOgNiTiOn iS ThE KeY
To bEiNg sUcCeSsFuL In cOlLeGe

Common Problem-Solving Strategies

- **Analogy** – Is this similar to another problem you've seen?
- **Brainstorming** – Consider steps to solve problem before jumping into code
  - Try to do an example "by hand" → outline steps
- **Solve sub-problems** – Is there a smaller part of the problem to solve?
- **Debugging** – Does your solution behave correctly?
  - What is it doing?
  - What do you expect it to do?
  - What area of your code controls that part of the output?
- **Iterative Development** – Can we start by solving a different problem that is easier?

Common Problem-Solving Strategies

- **Analogy** – Is this similar to another problem you've seen?
- **Brainstorming** – Consider steps to solve problem before jumping into code
  - Try to do an example "by hand" → outline steps
- **Solve sub-problems** – Is there a smaller part of the problem to solve?
- **Debugging** – Does your solution behave correctly?
  - What is it doing?
  - What do you expect it to do?
  - What area of your code controls that part of the output?
- **Iterative Development** – Can we start by solving a different problem that is easier?

Poll in with your answer!

What is the output produced by executing this code?

```
int a = 7;
int b = -1;
int c = 12;
if (a < b) {
    a *= 2;
} else if (b < a) {
    a /= 2;
} else {
    a = c;
}
if (c % 2 == 0) {
    c += 1;
}
if (b > 0) {
    b *= -1;
} else if (a < 0) {
    a *= -1;
}
System.out.println(a + " " + b + " " + c);
```

A. 7 -1 12

B. -3 -1 13

C. 3 -1 13

D. 12 1 12

E. -14 1 13