# CSE 121 Lesson 4: For Loops

Matt Wang

Spring 2024

| TAs: | | | | | | |
|---|---|---|---|---|---|---|
| | Andy | Anju | Archit | Arkita | Autumn | Christian |
| | Hannah H | Hannah S | Heather | Hibbah | Janvi | Jessie |
| | Jonus | Julia | Luke | Maria | Mia | Ritesh |
| | Shayna | Simon | Trey | Vidhi | Vivian | Gumball? |

**sli.do #cse121-4**

Today's playlist:
CSE 121 lecture beats 24sp

# Announcements & Reminders

- P0 due today!
- C1 releasing later today (due Tuesday, April 16th)
- Feedback for Creative Project 0 released yesterday!
- Resubmission Cycle 0 (R0) opening tomorrow (due Thursday, April 18th)
- Reminder: IPL is open!
  - in-person at MGH 334, uses a tool called “MyDigitalHand”
  - many hours: e.g. 12:30-9:30 today! (see schedule for more)
- Quiz 0: Thursday, Apr 25th during your quiz section.

# Resubmissions (or “resubs”)

Each week, you may resubmit one Programming Assignment or Creative Project with **no penalty**. The grade of your resubmission will completely replace your previous grade.

This is a huge opportunity: you get to resubmit your work after we grade it and give you feedback! Please take advantage of this :)

(if you miss an assignment and/or only finish it late – use a resub!)

# Resubmissions (or “resubs”)

Some logistics:

- there are 8 total resub cycles this quarter (and… 8 assignments…)
- an assignment is only eligible for resubmission for 3 cycles after feedback is released

To resubmit, you should:

- set the submission you want to be graded as “Final”
- **submit a Google form** to confirm your resubmission
  - you **must** fill out the form before the deadline for your resub to count!

# Last Time…

- Variables
  - Container that stores a specific data type
  - Must declare & initialize!
  - Manipulate, modify, reuse 
- Strings
  - Sequence of characters treated as one, yet can be indexed as individual parts
  - `char`, represents a single character

```
// declare AND initialize
int version = 5;
```

| L | a | u | f | e | y | ! |
|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 |

# Debugging – Live!

In P0, we asked you to do some debugging.

This is arguably the most important skill when programming – especially because programming is a social activity!

Let's do some live debugging :)

# (PCM) for loops!

For loops are our first *control structure*

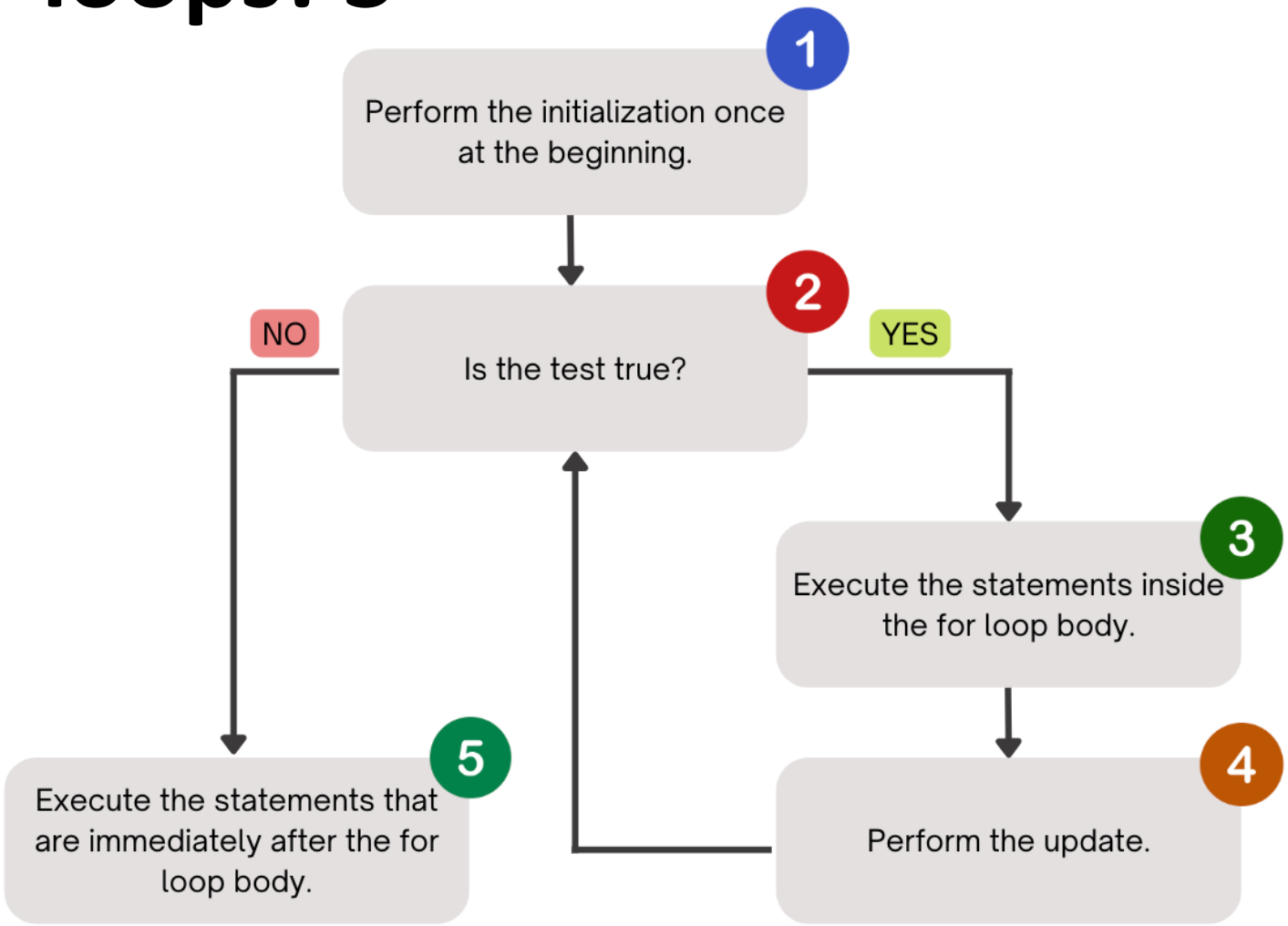
A syntactic structure that *controls* the execution of other statements.

```
for ( initialization ; test ; update ) {
    body (statements to be repeated)
}
```

# (PCM) for loops! 2

```
for (int counter = 1; counter <= 5; counter++) {
    System.out.println("I love CSE 121!");
}
```

# (PCM) for loops! 3



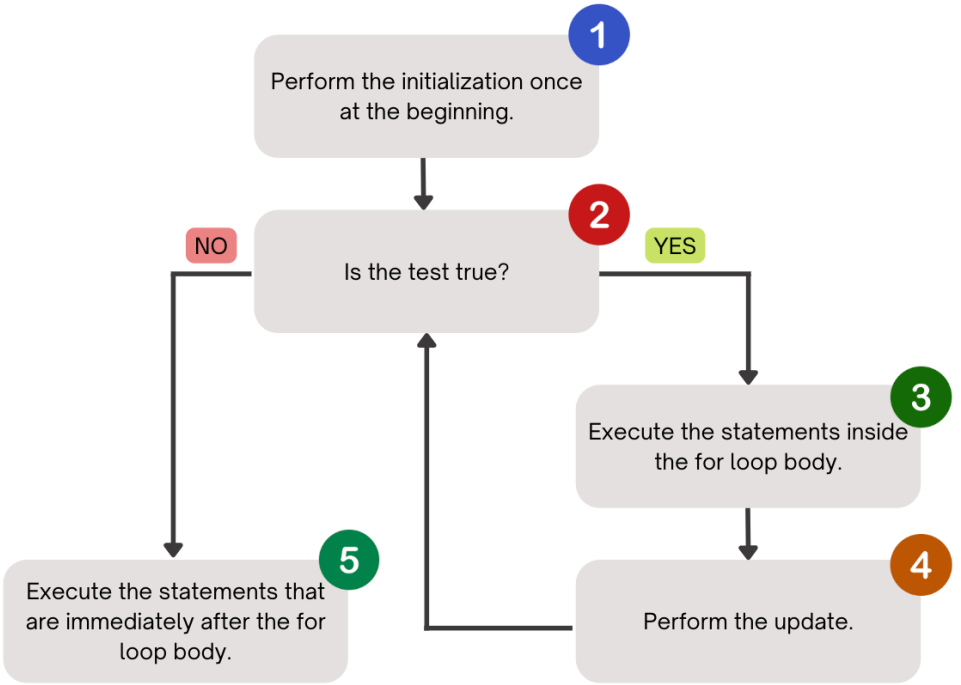
# (PCM) for loops! 4

counter

6

```
for (int counter = 1; counter <= 5; counter++) {
    System.out.println("I love CSE 121!");
}
```

```
I love CSE 121!
I love CSE 121!
I love CSE 121!
I love CSE 121!
I love CSE 121!
```



# Poll in with your answer!

sli.do #cse121-4

What output does the following code produce?

```
for (int i = 1; i <= 6; i++) {
  System.out.println(i + " squared = " + i * i);
}
```

## A.

```
i squared = i * i
i squared = i * i
i squared = i * i
i squared = i * i
i squared = i * i
i squared = i * i
```

## B.

```
i squared = i * i
i squared = i * i
i squared = i * i
i squared = i * i
i squared = i * i
i squared = i * i
i squared = i * i
```

## C.

```
1 squared = 1
2 squared = 4
3 squared = 9
4 squared = 16
5 squared = 25
6 squared = 36
```

## D.

```
1 squared = 1
2 squared = 4
3 squared = 9
4 squared = 16
5 squared = 25
6 squared = 36
7 squared = 49
```

# (PCM) String traversals

```
// For some String s
for (int i = 0; i < s.length(); i++) {
    // do something with s.charAt(i)
}
```

# Fencepost Pattern 1

Some task where one piece is repeated *n* times, and another piece is repeated *n-1* times and they alternate

**`h-u-s-k-i-e-s`**

# Fencepost Pattern 2

Some task where one piece is repeated *n* times, and another piece is repeated *n-1* times and they alternate