LEC 04

CSE 121

for Loops

BEFORE WE START

Talk to your neighbors:

What’s your favorite TV show right now?

Respond on sli.do!

Music: ❄ CSE 121 26wi Lecture Tunes ❄

Instructor:	Miya Natsuhara				
TAs:	Amogh	Hayden	Anum	Sam	Shayna
	William	Aki	Abdul	Ethan	Jesse
	Johnathan	Spencer	Janvi	Jessica	Minh
	Anant	Savannah	Navya	Paul	Cayden
	Reese	Tamsyn	Ruslana	Carson	

Questions during Class?

Raise hand or send here

sli.do #cse121

Announcements, Reminders

- Feedback for C0 released this morning!
  - Please view your feedback – crucial part of learning process
  - For regrades (not resubs), please make a private Ed post
  - Use the minimum grade calculator to track your grades!
- C1 releasing later today, due Tuesday, January 27th
- Quiz 0 is *next* Thursday, January 29th (in your registered quiz section)
  - Can't make it? Email Miya ASAP (mnats@cs.washington.edu)
- Resubmission Cycle 0 (R0) opening tomorrow, due Thursday January 29th

Reminder: 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!

Resub Logistics

Some logistics:

- There are 7* total resub cycles this quarter (and 8 assignments)
- Assignments eligible to resubmit for 3 cycles after feedback is out

To resubmit:

1. Make and submit your changes
2. Set the submission you want graded as “Final”
3. **Submit a Google Form**, with a reflection, to confirm your resub
   - You **must** submit the form before the deadline for resub to count

Practice: Think

sli.do #cse121

Suppose s contains the String "bubble gum".

Which statement would result in s containing "Gumball" instead?

b	u	b	b	l	e		g	u	m
0	1	2	3	4	5	6	7	8	9

A. `s.substring(7) + "ball";`

B. `s = s.substring(7, 9) + "ball";`

C. `s = s.charAt(7).toUpperCase() + "ball";`

D. `s = s.substring(7, 8).toUpperCase() + s.substring(8) + "ball";`

Practice: Think

sli.do #cse121

Suppose `s` contains the String `"bubble gum"`.

Which statement would result in `s` containing `"Gumball"` instead?

b	u	b	b	l	e		g	u	m
0	1	2	3	4	5	6	7	8	9

A. `s.substring(7) + "ball";`

B. `s = s.substring(7, 9) + "ball";`

C. `s = s.charAt(7).toUpperCase() +`
`"ball";`

D. `s = s.substring(7, 8).toUpperCase()`
`+ s.substring(8) + "ball";`

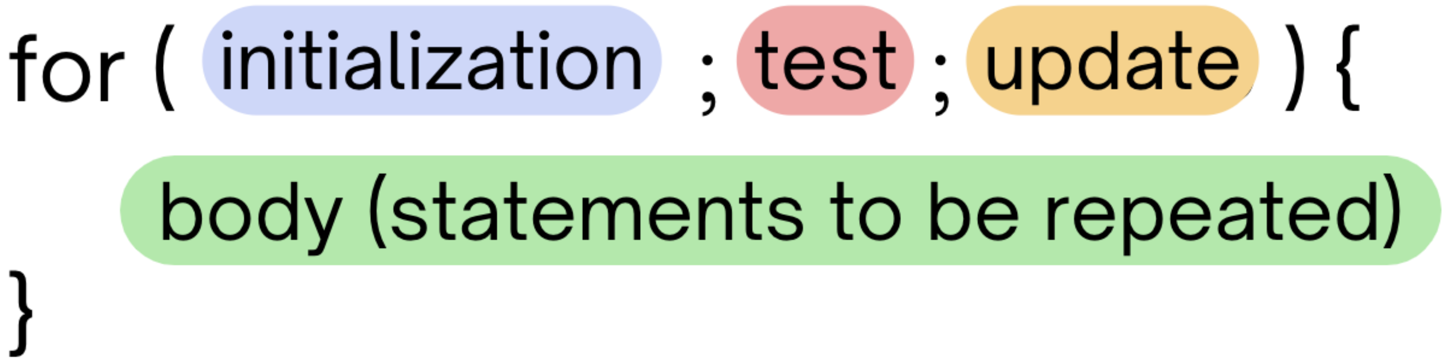
Chaining methods in expressions

b	u	b	b	l	e		g	u	m
0	1	2	3	4	5	6	7	8	9

```
s.substring(7, 8).toUpperCase() + s.substring(8) + "ball"

               "g".toUpperCase() + s.substring(8) + "ball"

                             "G" + s.substring(8) + "ball"

                             "G" +       "um"     + "ball"
```

PCM Review: for loops!

For loops are our first **control structure**: a syntax *structure* that *controls* the execution of other statements.

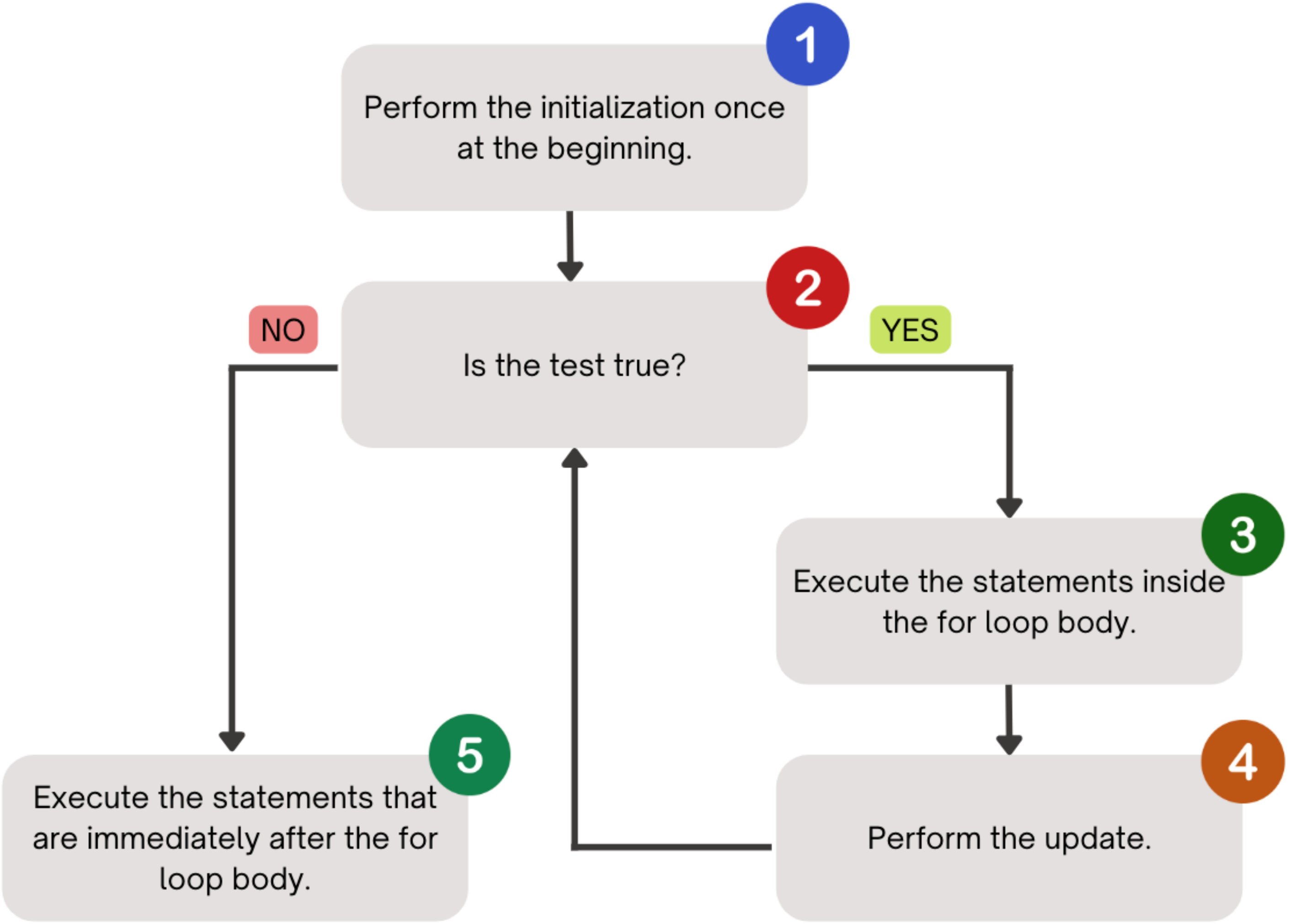


```
for ( initialization ; test ; update ) {
    body (statements to be repeated)
}
```

PCM Review: for loops (example)

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

PCM Review: for loops (a helpful flowchart)


1
Perform the initialization once at the beginning.
2
NO
Is the test true?
YES
3
Execute the statements inside the for loop body.
4
Perform the update.
5
Execute the statements that are immediately after the for loop body.

Practice: Think

sli.do #cse121

What output does the following code produce?

```
for (int i = 1; i <= 7; 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
```

Practice: Think

sli.do #cse121

What output does the following code produce?

```
for (int i = 1; i <= 7; 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 Review: String Traversals

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

b	u	b	b	l	e		g	u	m
0	1	2	3	4	5	6	7	8	9

Go Huskies?

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

The Fencepost Pattern

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`