COMP 250

Assignment 1 overview

Prepared by Héctor
hector.leosmendoza@mcgill.ca

Goals

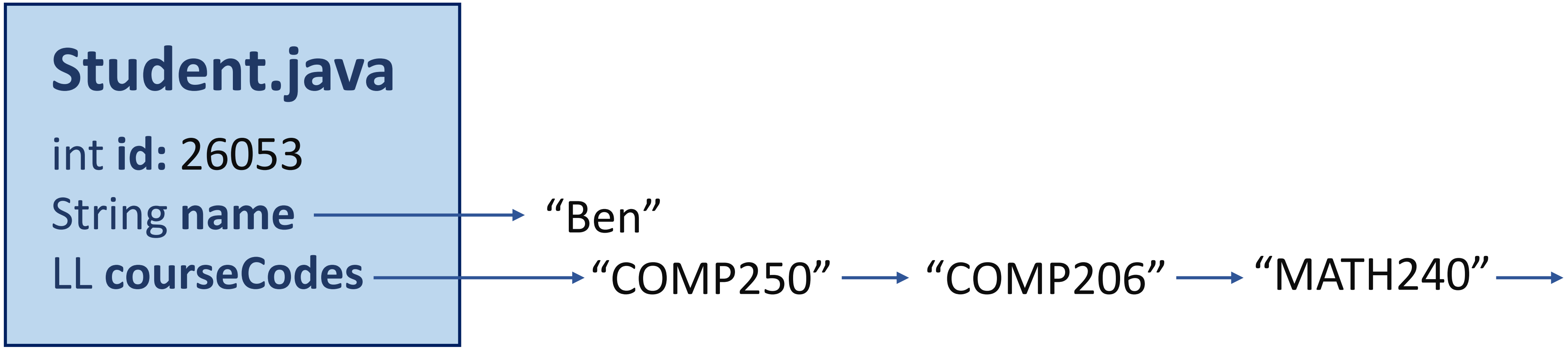
- Get you to work with arrays and Linked Lists
- Get you comfortable with Java / OOP
- Find a solution to a real life problem
- (Introduce you to hash maps)

Course.java

String **code**
int **capacity**
LL **studentTable**
int **size**
LL **waitlist**

Student.java

int **id**
String **name**
LL **courseCodes**



Student.java

int **id:** 26053

String **name** → “Ben”

LL **courseCodes** → “COMP250” → “COMP206” → “MATH240” →

Note: The official course registration information will be stored in the **Course** class.

Example

Course.java

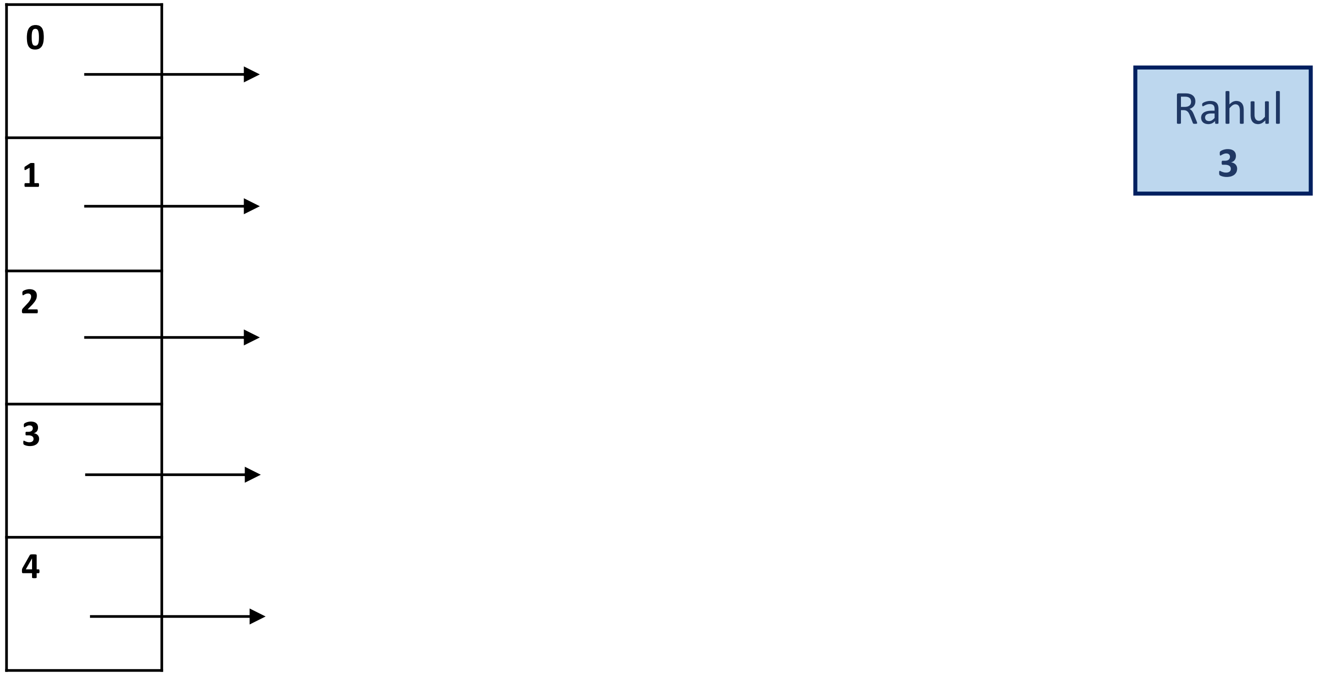
String **code:** “COMP250”

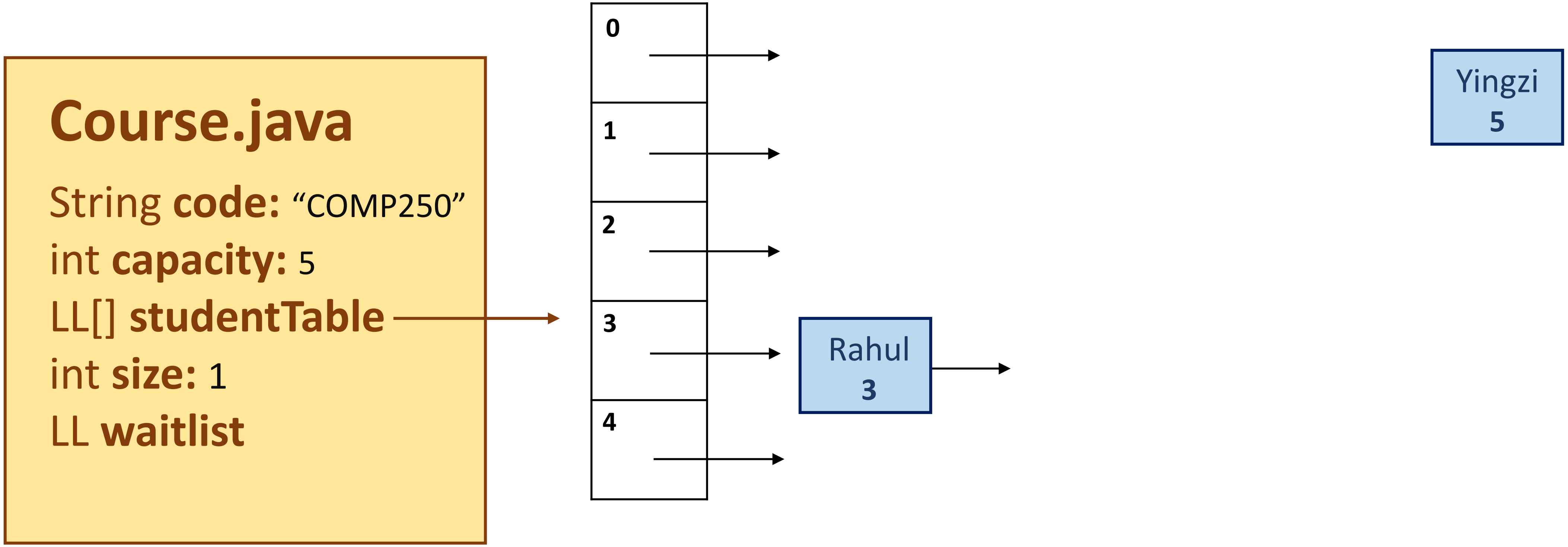
int **capacity:** 5

LL[] **studentTable**

int **size:** 0

LL **waitlist**





Course.java

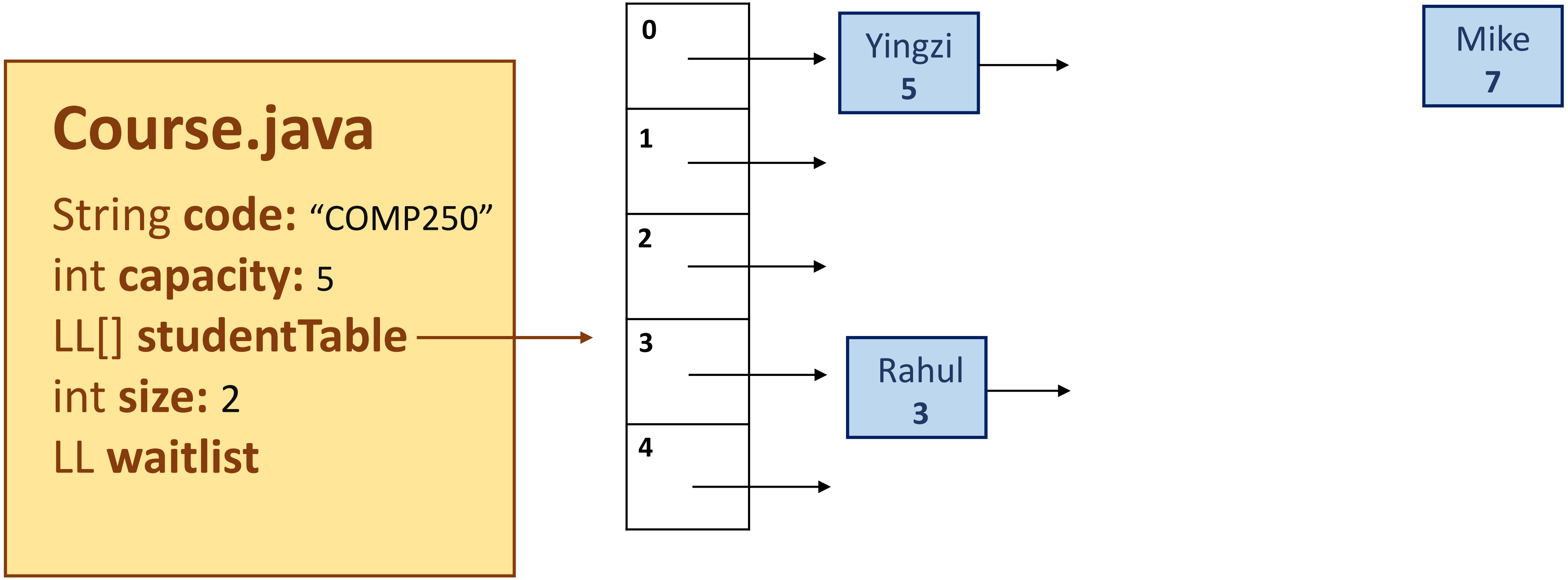
String **code:** "COMP250"

int **capacity:** 5

LL[] **studentTable**

int **size:** 1

LL **waitlist**



Course.java

String **code:** “COMP250”

int **capacity:** 5

LL[] **studentTable**

int **size:** 2

LL **waitlist**

0

1

2

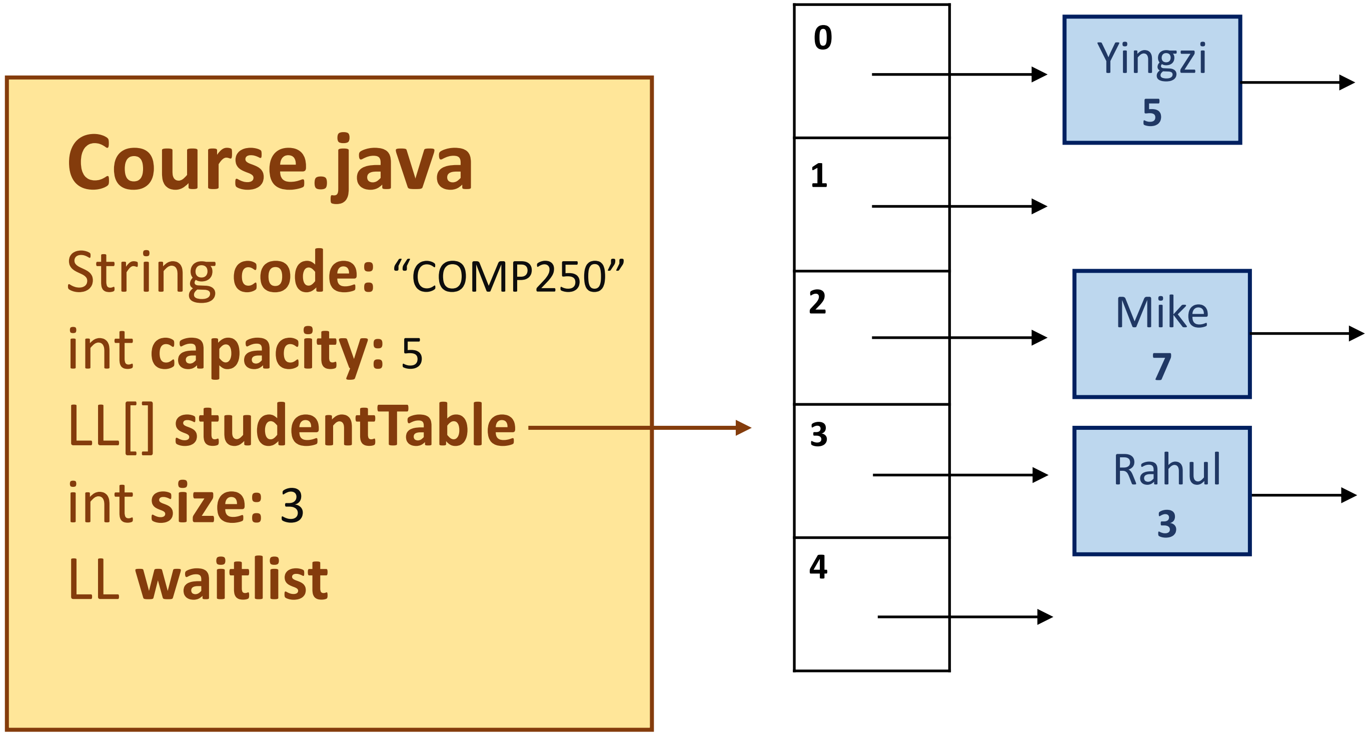
3

4

Yingzi
5

Rahul
3

Mike
7



Course.java

String **code:** “COMP250”

int **capacity:** 5

LL[] **studentTable**

int **size:** 3

LL **waitlist**

Course.java

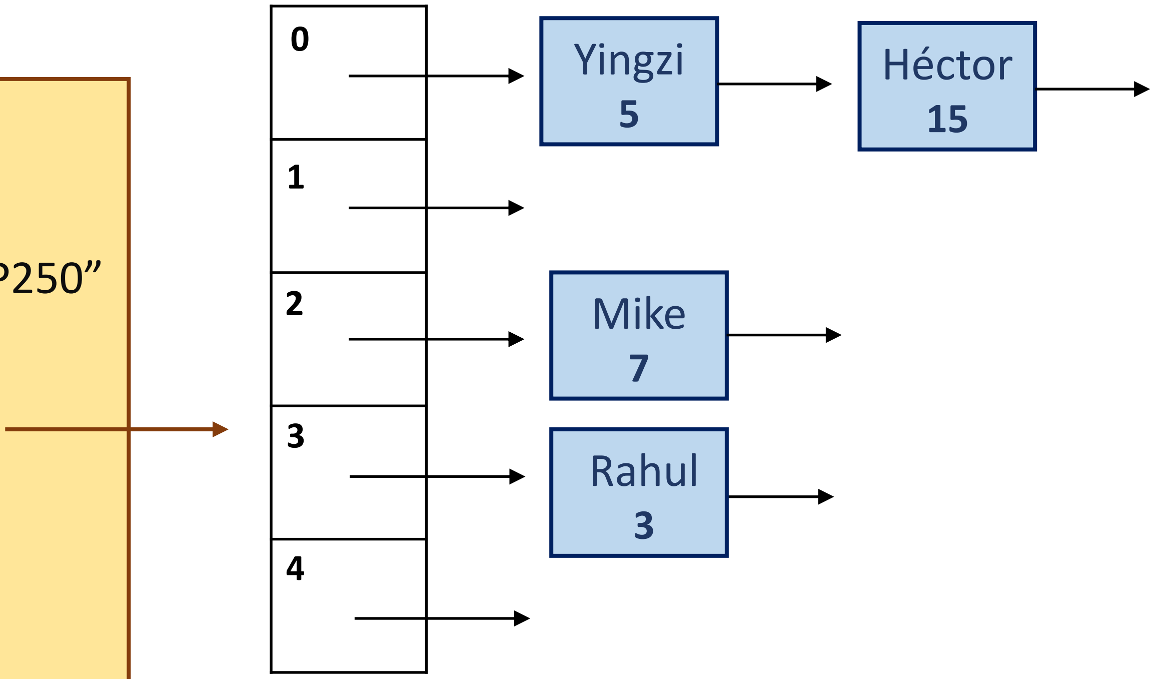
String **code:** “COMP250”

int **capacity:** 5

LL[] **studentTable**

int **size:** 4

LL **waitlist**



Course.java

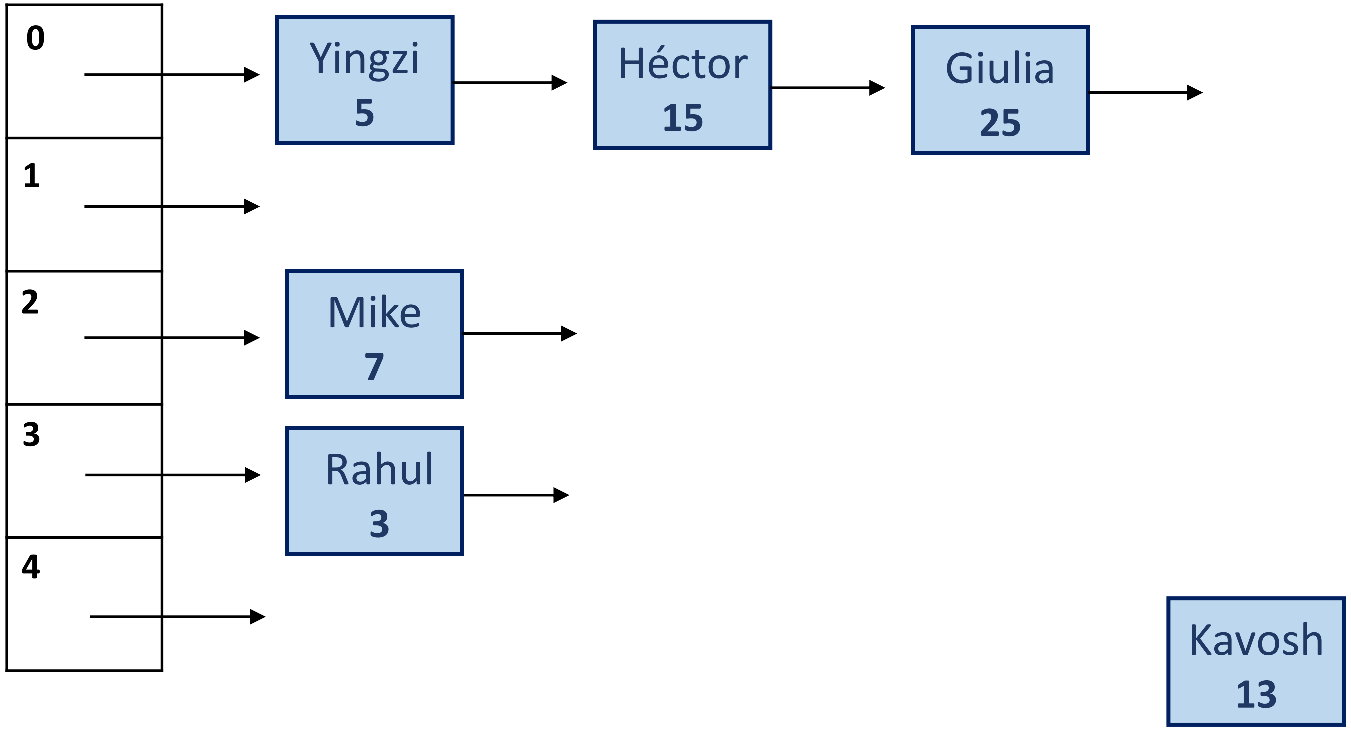
String **code:** “COMP250”

int **capacity:** 5

LL[] **studentTable**

int **size:** 5

LL **waitlist**



0

1

2

3

4

Yingzi
5

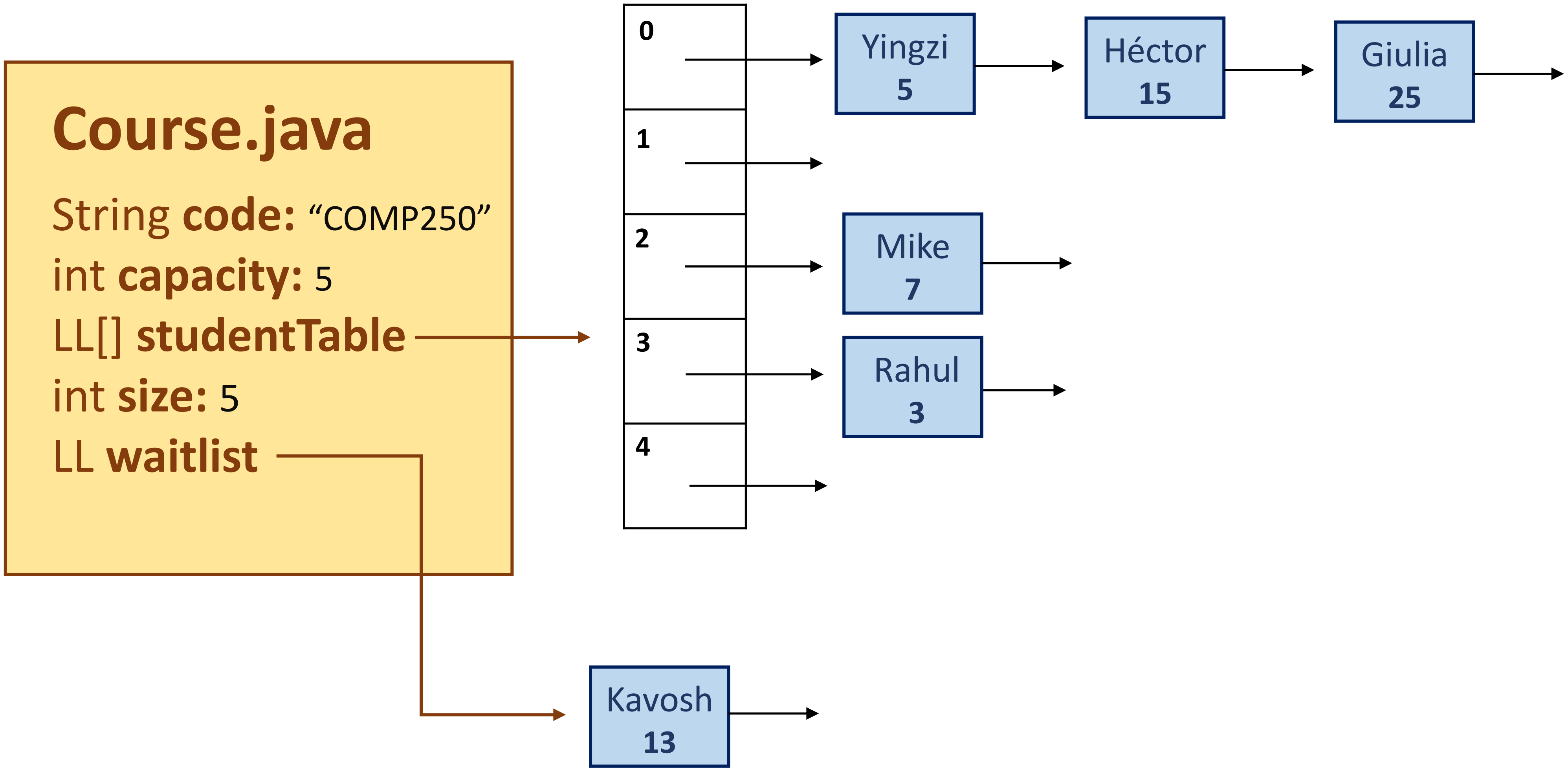
Héctor
15

Giulia
25

Mike
7

Rahul
3

Kavosh
13



Course.java

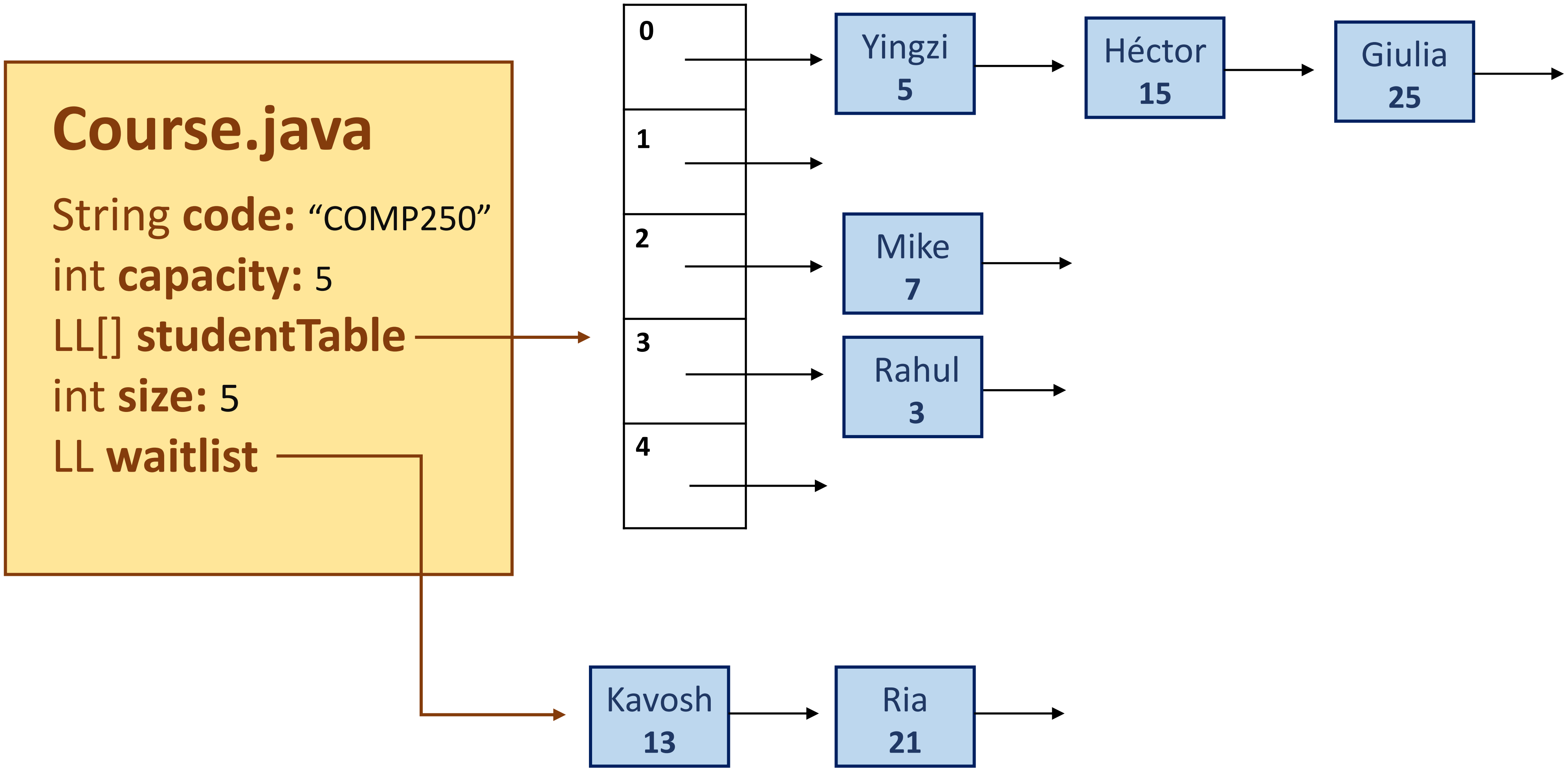
String **code:** “COMP250”

int **capacity:** 5

LL[] **studentTable**

int **size:** 5

LL **waitlist**



Course.java

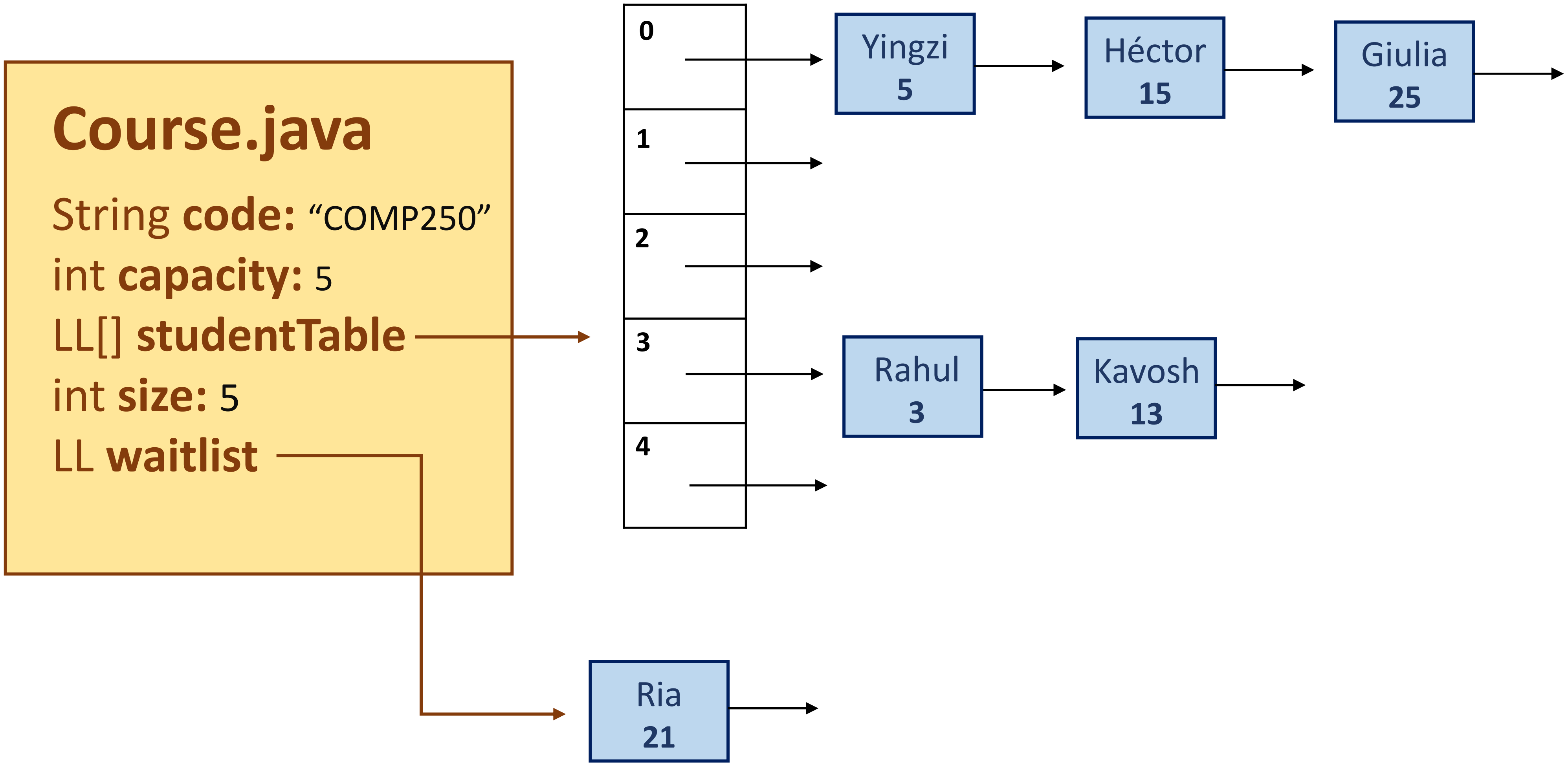
String **code:** “COMP250”

int **capacity:** 5

LL[] **studentTable**

int **size:** 5

LL **waitlist**



Course.java

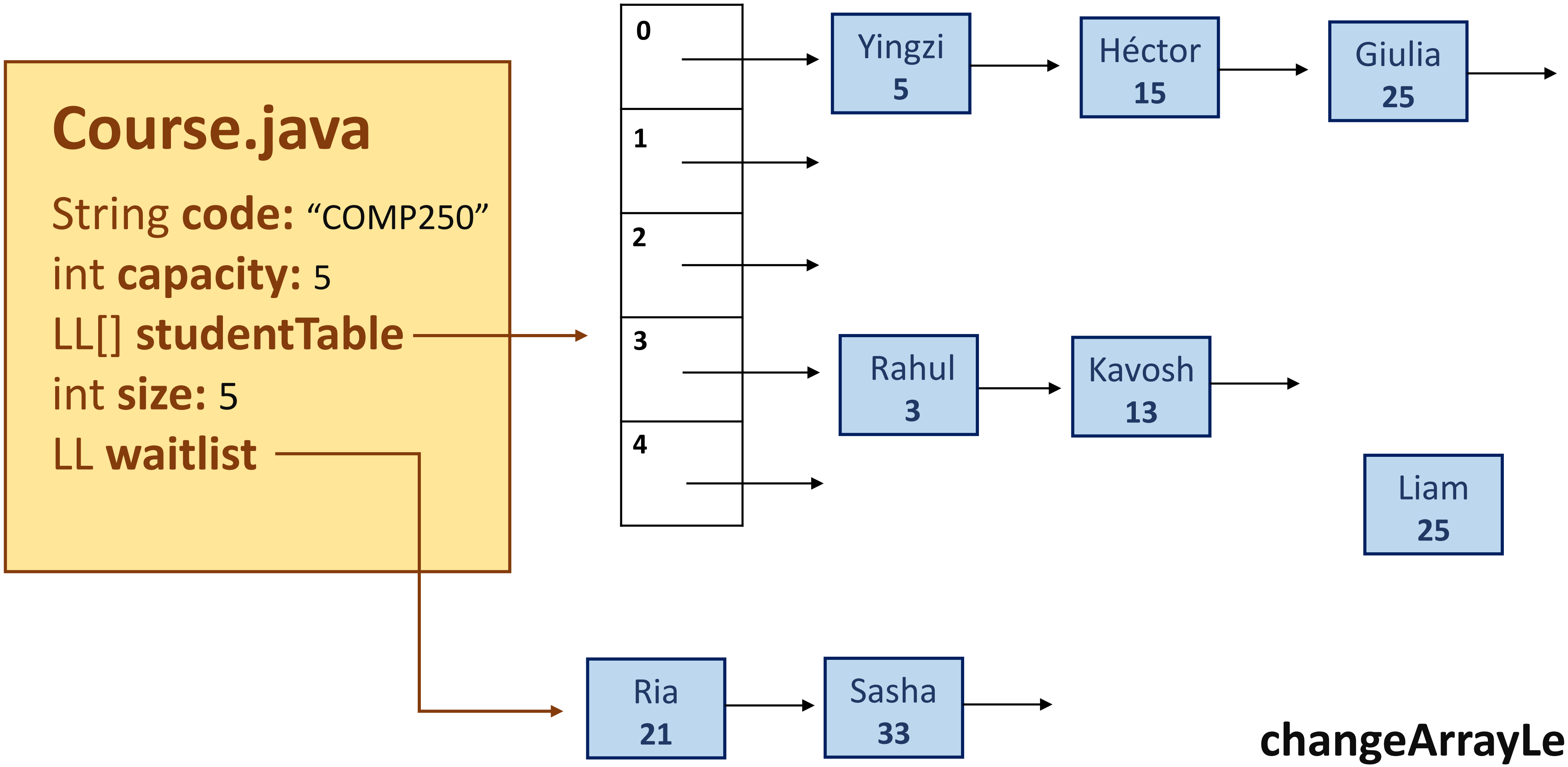
String **code:** “COMP250”

int **capacity:** 5

LL[] **studentTable**

int **size:** 5

LL **waitlist**



Course.java

String **code:** “COMP250”

int **capacity:** 5

LL[] **studentTable**

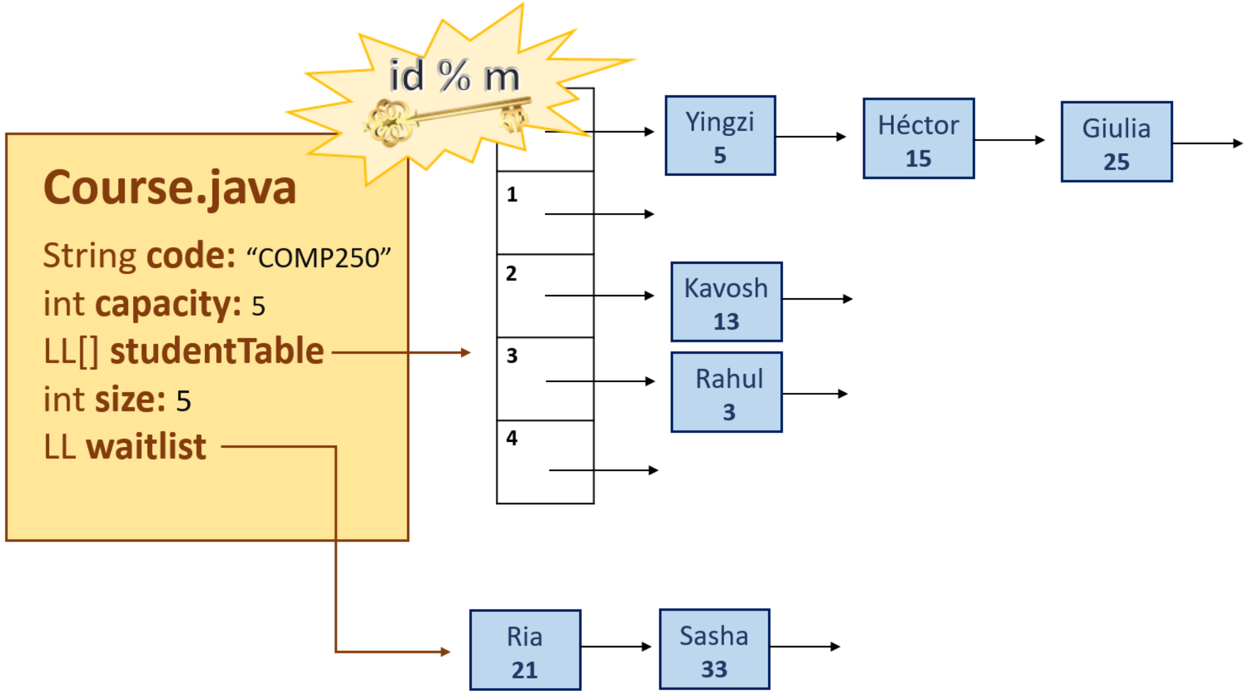
int **size:** 5

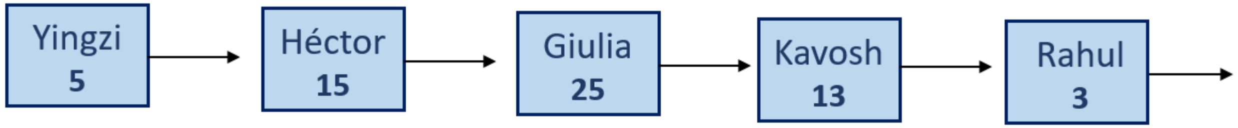
LL **waitlist**

changeArrayLength()

Why this data structure?

Which approach is faster at retrieving a Student based on its ID?

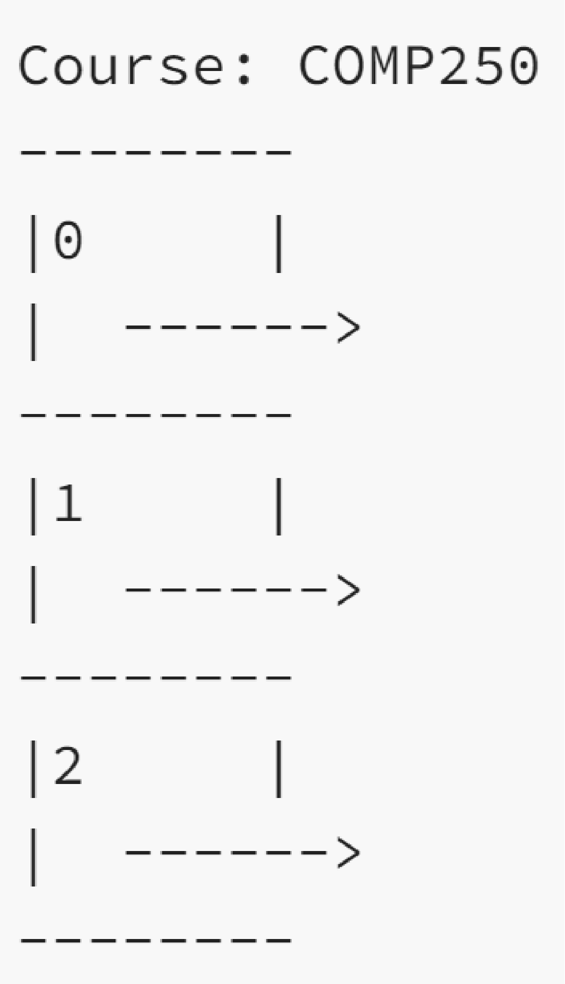




You'll learn more about **Hash Tables** later on…

toString()

```java
Course comp250 = new Course("COMP250", 3);
System.out.println(comp250);
```



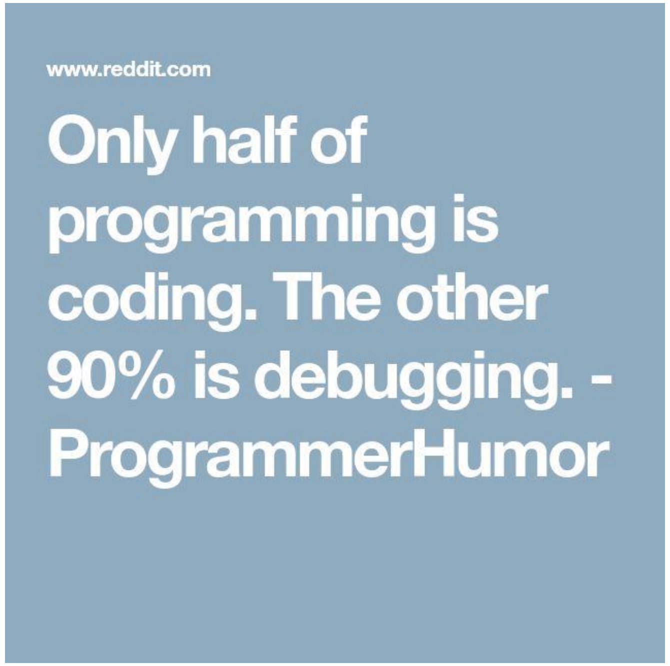
```
Course: COMP250
--------
|0     |
|  ------>
--------
|1     |
|  ------>
--------
|2     |
|  ------>
--------
```

Last pieces of advice…

- Make sure to understand how SLinkedList works before using it.
- Remember to update the Student’s courseCodes as you add or remove a student from a course.
- Don’t forget to initialize the studentTable’s slot with a linked list when you’re adding a student there for the first time.
- When resizing the array, don’t forget to properly update this.studentTable.
- Print statements are your best friends (but the debugger is also really useful)
- Test your code extensively and use student tests
- Make sure to set up JUnit in your IDE before running the test classes (you don’t have to do any changes to those files)
- Start early on and come to OH!

Grading

- Exposed tests and private tests
- Why private tests?
    - That’s how debugging works in real life, you don’t know which method might have an error



www.reddit.com

Only half of programming is coding. The other 90% is debugging. - ProgrammerHumor

https://www.pinterest.ca/pin/508484614172225130/

Good luck! Any questions?

Any questions about the slides please contact me at hector.leosmendoza@mcgill.ca. General questions about the assignment should be posted on Ed.