CSE 121 Lesson 15:
Arrays & Reference Semantics

Elba Garza & Matt Wang

Winter 2024

TAs:  Abby    Aishah    Anju    Annie    Archit    Ayesha    Christian
      Hannah  Heather  Hibbah  Jacob  James  Janvi   Jasmine
      Jonus   Julia    Lucas   Luke   Maria   Nicole  Shananda
      Shayna  Trey     Vidhi   Vivian

sli.do #CSE121-15

Today’s playlist:
CSE 121 24wi lecture beats 😊
Reminders & Announcements

• C3 released Wednesday, due Tuesday Feb 27th
• R5 released yesterday, due Thursday Feb 29th
  • note: last chance for P1!
• Quiz 2 next Thursday, Feb 29th
  • topics: everything up until Arrays on Wednesday (i.e. not today’s material)
  • next week: many practice quizzes :)
• In the future: Final Exam (Tuesday March 12th at 12:30 – 2:20pm)
  • more logistical details coming soon!
What would the array `a` store at the end of this `arrayMystery` method if 
`{-20, 20, 26, 32, 50, 3}` was passed in?

```java
public static void arrayMystery(int[] a) {
    for (int i = a.length - 1; i >= 1; i--) {
        if (a[i] > a[i - 1] + 10) {
            a[i - 1] = a[i - 1] + 5;
        }
    }
}
```

A. `{-20, 20, 26, 32, 50, 3}`
B. `{-15, 25, 31, 37, 55, 8}`
C. `{-15, 25, 31, 37, 50, 3}`
D. `{-15, 20, 26, 37, 50, 3}`
Value Semantics vs. Reference Semantics

- Applies when working with primitive types
- Variables/parameters hold a *copy* of the actual value

- Applies when working with objects
- Variables/parameters hold a *reference* to the object

```java
def age = 10;

age
```

```java
def arr = new int[4];

arr
```
(PCM) Value Semantics vs. Reference Semantics

```java
int a = 3;
int b = a;
a = 99;
```

```java
int[] list1 = {4, 8, 15, 16, 23};
int[] list2 = list1;
list1[1] = 99;
```

```
age = 10;
```

```
int[] arr = new int[4];
name: arr (int[])
```

Lesson 15 - Winter 2024
Without knowing what `someMethod` does, what are the possible values of `num`?

```java
int num = 42;
someMethod(num);
System.out.println(num);
```

A. anything!
B. just 42
Poll in with your answer!

Without knowing what anotherMethod does, what are the possible values of nums[0]?

```java
int[] nums = {42, 43, 44};
anotherMethod(nums);
System.out.println(nums[0]);
```

A. anything!
B. just 42
boolean test = true;
flipValue(test);
public static void flipValue(boolean b) {
    b = !b;
}

boolean[] tests =
    {true, true, false, true, false, false};
flipValues(tests);
public static void flipValues(boolean[] b) {
    for (int i = 0; i < b.length; i++) {
        b[i] = !b[i];
    }
}
null

The absence of a reference!
Sort of like a "zero-equivalent" for references!
Default value for object types

NullPointerExceptions occur when you try to ask a reference “do something”, but there isn't a reference there...it's null!
"do something" such as get an element from an array, or call a method on an object)
Four lines of output would be produced by this code. What would those four lines be?

```java
public static void main(String[] args) {
    int x = 0;
    int[] a = new int[4];

    x++;
    mystery(x, a);
    System.out.println(x + " " + Arrays.toString(a));

    x++;
    mystery(x, a);
    System.out.println(x + " " + Arrays.toString(a));
}

public static void mystery(int x, int[] a) {
    x++;
    a[x]++;
    System.out.println(x + " " + Arrays.toString(a));
}
```