CSE 11
Accelerated Intro to Programming
Discussion Section 6

Shihua Lu, Spring 2021

This discussion is being recorded

Logistics

- PA5 due today at 11:59PM
- PA6 Released Due next wednesday
- Exam this week - Thursday 8am - Sunday 8am (PST)

No collaboration

Open book

Start early

For-in loops For-each loops

- run the body of their loops once per element in a given array
- follows the order that they appear in the array, with the "loop variable" assigned to that element
- Their syntax looks like:

```
for (/* element type */ /* variable name */: /* array */) {
    /* loop body */
}
```

Example :

```
String[] messages = {"Hello", "CSE11"};


for (String message: messages) {
    System.out.println(message);
  }
```

for each String message in the array messages

prints

```
Hello

CSE11
```

Counted for loops

More powerful than for-in loops

- **More fine grained control compared to for-in loops - these are not limited to running once per element in an array**
- **Their syntax looks like:**

```
for (/* initializer */; /* condition */; /* update */) {
    /* body */
  }
```

initialize a variable

must evaluate to a boolean

updating the variable

```
                   ------------------------------------------
                   |                                        |
                   v                                        |
---------------    -------------          --------    ----------    |
| initializer | -> | condition | -------> | body | -> | update | -
---------------    -------------   true   --------    ----------
                         |
                   false |
                         |
                         v
                       (exit)
```

```
int[] arr = {1, 2, 3};
for (int ele : arr) {
    ele = 2;
}
```

X

```
for (int i = 0;
        i < arr.length;
            i += 1) {
    arr[i] += 1;
}
```

Counted for loops

- **body of counted for loops contain other can contain multiple Java statements, including multiple method calls, variable definitions and updates, if statements, and even other loops**

Nested for-loops

```
// Start with the top header of the table
String result = "\t1\t2\t3\t4\t5\n";

for (int i = 1; i <= 5; i += 1) {
    // Print the left header
    result += i;

    // Print the row
    for (int j = 1; j <= 5; j += 1) {
        result += "\t" + (i * j);
    }

    // Add a newline to finish the row
    result += "\n";
}

System.out.println(result);
```

Prints the times table from 1 to 5 to the terminal:

j

i

```
        1     2     3     4     5
1       1     2     3     4     5
2       2     4     6     8     10
3       3     6     9     12    15
4       4     8     12    16    20
5       5     10    15    20    25
```

i = 1

i = 2

i = 3

i = 4

i = 5

PA6

- https://github.com/CSE11-SP21-Assignments/cse-11-sp21-pa6-starter

Exam

Programming assignment: Coding + Video → Reserve enough time

Materials: Review lectures / readings / quizzes / PAs / discussions

- Class
  - abstract classes
  - interfaces
  - normal classes
- fields
  - types
    - primitive types
      - int
      - double
      - boolean
    - classes
      - String
      - Pair
      - Point
      - ...
  - objects (new)
- methods
  - constructors
  - general method (Returntype functionName (ArgType arg1, ...) { })
  - main method

tester (t.checkExpect)

import tester.*; (./run)

if / else if / else

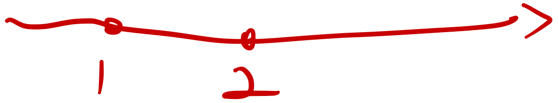
for-loops

String methods

Math methods

Not guaranteed to be complete

"a"   "abc"   1   2



"a".compareTo("abc"); → negative

1 - 2 → negative

Thanks!