Wrap-Up

EECS1021:
Object Oriented Programming:
from Sensors to Actuators
Winter 2019

CHEN-WEI WANG

Why this Course?

- *Computational thinking (CT)* is a fundamental skill for **everyone**, not just for computer scientists.
    - Reference: Wing, J.M., 2006. *Computational thinking*. Communications of the ACM, 49(3), pp.33 – 35.
    - Thinking like a computer scientist means **more than being able to program** a computer. It requires **thinking at multiple levels of abstraction**.
        - ***Level of Java Code***: How Programs Behave at Runtime
        - ***Above the Level of Code***: *Logical rationale* behind some *functioning*/*malfunctioning* code.
- Being able to think *abstractly* without seeing changes on a physical device is an important skill you are expected to acquire when graduating.
    - Think of programming interviews at Google: Given problems described in English, solve it on a whiteboard.

What You Learned (1)

- *Procedural Programming in Java*
  - primitive data types
  - assignments
  - casting vs. coercion for numbers
  - Boolean expressions, logical operators, short-circuit evaluation
  - `if`-statements
  - Solving problems *iteratively*: `for` vs. `while` loops
  - one-dimensional arrays

What You Learned (2)

- *Object-Oriented Programming in Java*
  - classes, attributes, objects, reference data types
  - methods: constructors, accessors, mutators, helper
  - dot notation, context objects, method calls
  - aliasing
  - Java API: `Math`, `Scanner`, `ArrayList`, `Hashtable`
- keywords: `final`, `this`, `static`

What You Learned (3)

- *Integrated Development Environment (IDE) for Java: Eclipse*
  - Compile Time vs. Runtime
    - Syntax Errors
    - Type Errors
    - Logical Errors
  - Creating Console App's via Classes with `main` method
  - User interactions
  - *Breakpoints and Debugger*

Beyond this course. . .

- Java Tutorials

  https://www.youtube.com/playlist?list=PL5dxAmCmjv_5NRNPG3OiWZWAqmvCjiLfG

- Two-Dimensional Arrays

  https://www.eecs.yorku.ca/~jackie/teaching/lectures/index.html#EECS1022_W18

- Advanced Object-Oriented Programming

  https://www.eecs.yorku.ca/~jackie/teaching/lectures/index.html#EECS2030_F18

Wish You the Best

- What you have learned will be **assumed** in `EECS2030`.
- Do **not** abandon Java during the break!!

Course Evaluation

`courseevaluations.yorku.ca`