Wrap-Up

EECS1022 Sections M & N:
Programming for Mobile Computing
Winter 2021

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
    - VARIABLE ASSIGNMENTS
    - NUMERICAL CASTING VS. COERCION
    - BOOLEAN EXPRESSIONS, LOGICAL OPERATORS
    - SHORT-CIRCUIT EVALUATION
    - CONDITIONALS
    - LOOPS
    - ONE-DIMENSIONAL ARRAYS
    - TWO-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
  - Reference Aliasing
  - Java API: `Math`, `Scanner`, `ArrayList`, `Hashtable`
- KEYWORDS: `final`, `this`, `static`

What You Learned (3)

- **Integrated Development Environment (IDE): Eclipse**
  - Compile Time vs. Runtime
    - Syntax Errors
    - Type Errors
    - Logical Errors
  - Console App & `main` Method
  - Mobile App
  - JUnit Tests & Assertions
  - **Breakpoints & Debugger**

Beyond this course. . . (1)

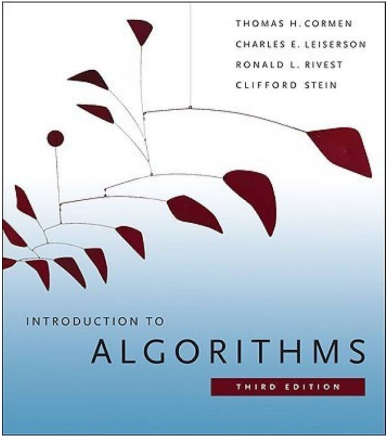
- Advanced Object-Oriented Programming

  `https://www.eecs.yorku.ca/~jackie/teaching/lectures/index.html#EECS2030_F19`

- Lots of Coding Interview Problems

  `https://leetcode.com/`

Beyond this course. . . (2)



- *Introduction to Algorithms (3rd Ed.)* by Cormen, *etc.*
- DS by DS, Algo. by Algo.:
  - *Understand* math analysis
  - *Read* pseudo code
  - *Translate* into Java code
  - *Write and pass* JUnit tests

Wish You the Best

- What you have learned will be ***assumed*** in `EECS2030`.
- Do ***not*** abandon Java during the break!!
- When we return to campus, come by and say hi ☺

Course Evaluation

`courseevaluations.yorku.ca`