Wrap-Up

EECS2030 B: Advanced Object Oriented Programming
Fall 2018

CHEN-WEI WANG

What You Learned (1)

- *Procedural Programming in Java*
  - Exceptions
  - Recursion (implementation, running time, correctness)
- *Data Structures*
  - Arrays
  - Maps and Hash Tables

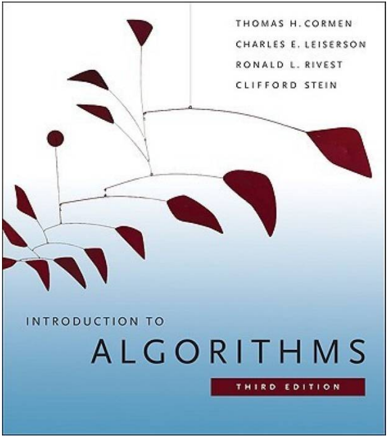
What You Learned (2)

- *Object-Oriented Programming in Java*
  - classes, attributes, encapsulation, objects, reference data types
  - methods: constructors, accessors, mutators, helper
  - dot notation, context objects
  - aliasing
  - inheritance:
    - code reuse
    - expectations
    - static vs. dynamic types
    - rules of substitutions
    - casts and `instanceof` checks
    - polymorphism and method arguments/return values
    - method overriding and dynamic binding: e.g., `equals`
    - abstract classes vs. interfaces
    - generics (vs. collection of `Object`) [Optional]

What You Learned (3)

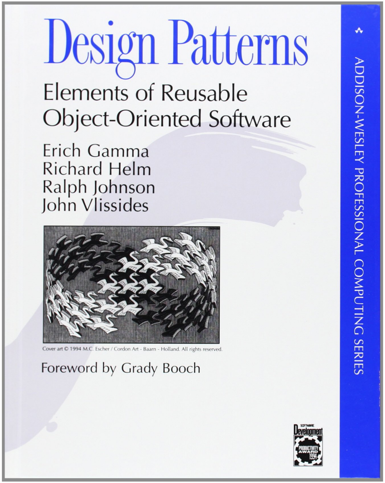
- *Integrated Development Environment (IDE) for Java: Eclipse*
  - Break Point and Debugger
  - Unit Testing using JUnit

Beyond this course. . . (1)



- *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

Beyond this course. . . (2)


Design Patterns
Elements of Reusable Object-Oriented Software
Erich Gamma
Richard Helm
Ralph Johnson
John Vlissides
Cover art © 1994 M.C. Escher / Cordon Art - Baarn - Holland. All rights reserved.
Foreword by Grady Booch
ADDISON-WESLEY PROFESSIONAL COMPUTING SERIES
Software Development Productivity Award 1994

- *Design Patterns: Elements of Reusable Object-Oriented Software* by Gamma, *etc.*
- Pattern by Pattern:
  - *Understand* the problem
  - *Read* the solution (not in Java)
  - *Translate* into Java code
  - *Write and pass* JUnit tests

Beyond this course. . . (3)

Visit my lectures on *EECS3311 Software Design*:

`http://www.eecs.yorku.ca/~jackie/teaching/lectures/index.html#EECS3311_F18`

- Design by Contracts
- Design Patterns
- Program Verification

Wish You All the Best

- What you have learned will be **assumed** in `EECS2011`.
- Logic is your friend: Learn/Review `EECS1019/EECS1090`.
- Do **not** abandon Java during the break!!
- Feel free to get in touch and let me know how you're doing :D