

Wrap-Up



EECS2030 B & G: Advanced
Object Oriented Programming
Fall 2025

CHEN-WEI WANG

What You Learned (1)

- ***Procedural Programming in Java***
 - Exceptions
 - Recursion (thinking, implementation, tracing)
- ***Data Structures***
 - Arrays

What You Learned (2)

- ***Object-Oriented Programming in Java***
 - classes, attributes, objects, reference data types
 - methods: constructors, accessors, mutators, helpers
 - dot notation, context objects
 - aliasing
 - inheritance:
 - code reuse, single-choice principle, cohesion
 - expectations
 - rules of substitutions
 - static vs. dynamic types
 - polymorphism, dynamic binding
 - polymorphic method parameters
 - polymorphic collections
 - polymorphic method return types
 - compilable casts, `ClassCastException`, `instanceof` checks
 - method overriding and dynamic binding: e.g., `equals`

What You Learned (3)

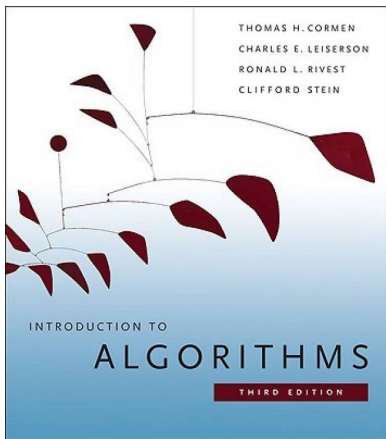
- ***Integrated Development Environment (IDE): Eclipse***
 - Break Point and Debugger
 - Unit Testing using JUnit
 - Test Driven Development (TDD), Regression Testing

[https://www.eecs.yorku.ca/~jackie/teaching/
lectures/index.html#EECS2030_F21](https://www.eecs.yorku.ca/~jackie/teaching/lectures/index.html#EECS2030_F21)

- ***Generics***

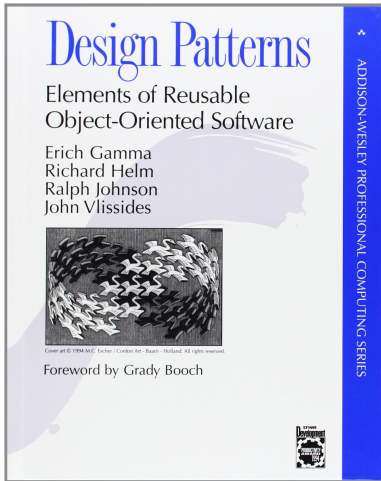
[Week 10 & 11]

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* by Gamma, etc.
- Pattern by Pattern:
 - **Understand** the problem
 - **Read** the solution (not in Java)
 - **Translate** into Java code
 - **Write and pass** JUnit tests

Wish You All the Best

- What you have learned will be **assumed** in EECS2101.
- 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