CS2312 Problem Solving and Programming

2023-2024 Semester A

Department of Computer Science, City University of Hong Kong

Instructor: Dr. Helena WONG

Your Attention, Please!

- Not a pure programming course.
- Your first programming course and your coming software design course **has a large gap** in terms of the level of abstraction required.
- This course is to help you to raise the level of abstraction from pure programming to **a logical organization of software code** based on the requirements of the targeted applications to be developed.

[Borrowed from Dr. Ricky CHAN's notes in Spring 2013]

Related Courses:
CS2310 Computer Programming
CS2312 Problem Solving and Programming
CS3342 Software Design
CS3343 Software Engineering Practice

Java Programming and OO

- [Teaching Focus #1] Java Programming
  - Crash introduction of basics

    you have learnt C++ already, we can move fast onto java
  - Intensive study of key and advanced techniques

    target: pave the way for Part 2

- [Teaching Focus #2] Doing the OO
  - Object Oriented - concepts/design/principles/practices

Intended Learning Outcomes - Briefly:

1. **[OO] Understand OO concepts**
2. **[OOD] Design OO solutions**
3. **[OOP] Implement the OO solutions in Java**
4. **[Practices] Apply the best practices in Java programming**
5. **[Review] Evaluate and review OO design and code**

Python and Functional Programming

- [Teaching Focus #3]

  Python and Functional Programming

  - Given in week 12-13

Textbook and Materials

- [Focus #1] Java Programming

  Textbook: C.S. Horstmann, and G. Cornell, **Core Java^{TM} Volume I**, Prentice Hall.

  Other books on my desk:

  - Walter Savitch, **Absolute Java**, Addison-Wesley.
  - Y. D. Liang, **Intro. to Java^{TM} Programming Comprehensive Version**, Pearson.

  Official site of Java, tutorial: http://docs.oracle.com/javase/tutorial/index.html

- [Focus #2] OO concepts/design/principles/practices

  - Materials from Dr. Sam NG for his teaching of a previous course: CS2332 OOP $^{\text{in C++}}$
    Sam is also the author of the current syllabus of CS2312.
  - Materials from Dr. Ricky CHAN $^{\text{[CS2312 / Spring 2013, CS3342]}}$, Dr. Jacky KEUNG $^{\text{[CS3342]}}$
  - More.. [Check out at our courseweb]

- **Acknowledgments:**

  "Some of the material for this course was influenced by and, in some cases, directly borrowed from, materials available on the web for similar courses at other universities. I thank the instructors who posted their materials on the web." [Borrowed from http://www.cse.ohio-state.edu/~neelam/courses/45923/]

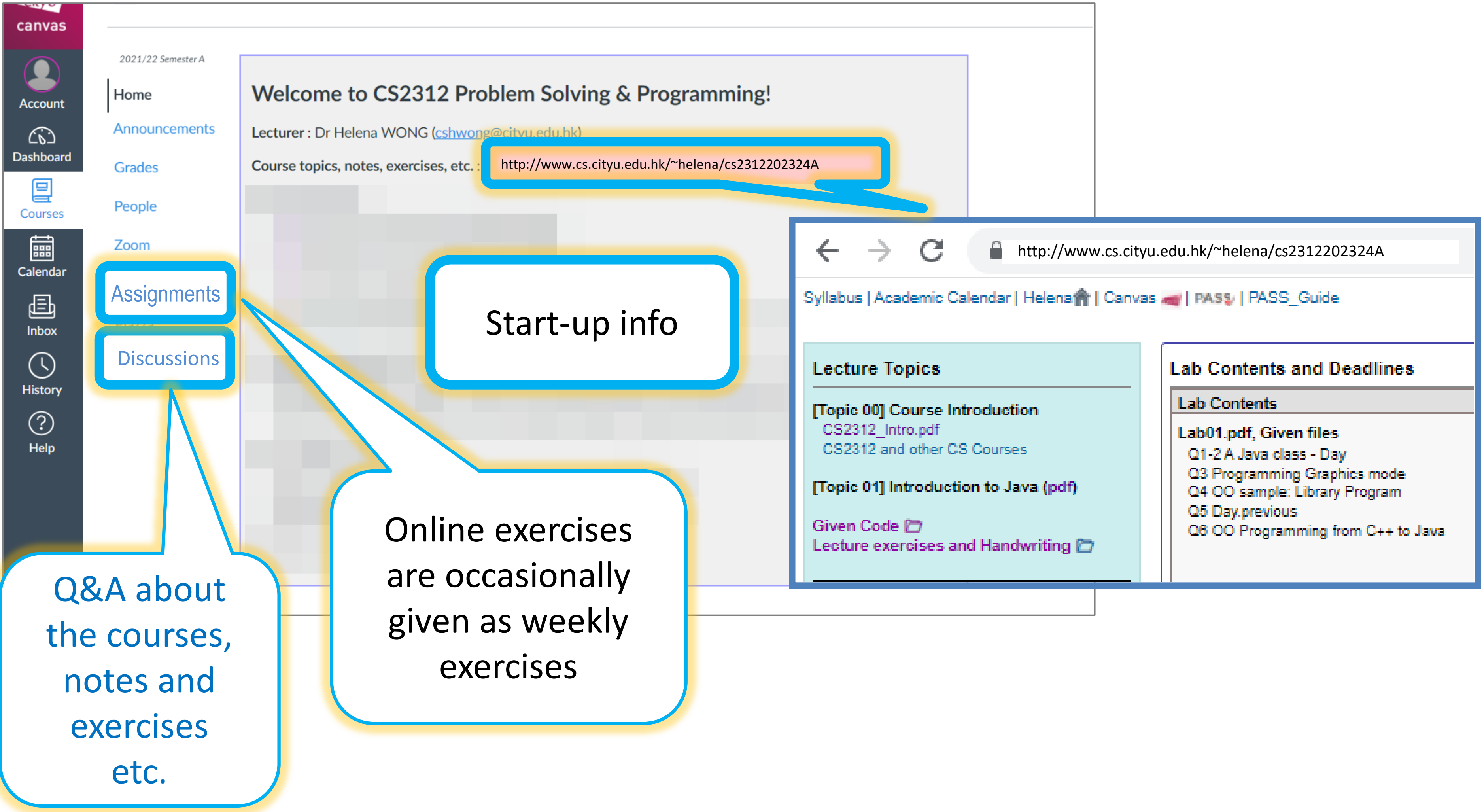
Tentative Assessment Pattern

- **Coursework: 50%**
  - **[10 %] Progress assessments**
    - Based on Weekly Progress Assessments
  - **[5 %] Short Quiz (Week 5 lecture 12:00)**
  - **[20 %] Midterm (Week 12 Saturday 8pm)**
  - **[15 %] Programming assignment**
  - **+[0-5%] Special discretion** (continuous attendance, observed effort, quiz-redo deadline given)
    - Just to save marginal cases ; *NOT a kind of bonus*
    - Will consider after marking of the exam paper; apply only to those who will get C or below
- **Exam: 50%**

Passing Criteria:

At least 30% of the maximum mark for the examination must be obtained; and
At least 35% of the maximum mark for the overall final mark must be obtained

Course Web



canvas

Account

Dashboard

Courses

Calendar

Inbox

History

Help

2021/22 Semester A

Home

Announcements

Grades

People

Zoom

Assignments

Discussions

Welcome to CS2312 Problem Solving & Programming!

Lecturer : Dr Helena WONG (cshwong@cityu.edu.hk)

Course topics, notes, exercises, etc. : http://www.cs.cityu.edu.hk/~helena/cs2312202324A

[redacted]

http://www.cs.cityu.edu.hk/~helena/cs2312202324A

Syllabus | Academic Calendar | Helena | Canvas | PASS | PASS_Guide

Lecture Topics

[Topic 00] Course Introduction
CS2312_Intro.pdf
CS2312 and other CS Courses

[Topic 01] Introduction to Java (pdf)

Given Code

Lecture exercises and Handwriting

Lab Contents and Deadlines

Lab Contents

Lab01.pdf, Given files
Q1-2 A Java class - Day
Q3 Programming Graphics mode
Q4 OO sample: Library Program
Q5 Day.previous
Q6 OO Programming from C++ to Java

Start-up info

Online exercises are occasionally given as weekly exercises

Q&A about the courses, notes and exercises etc.

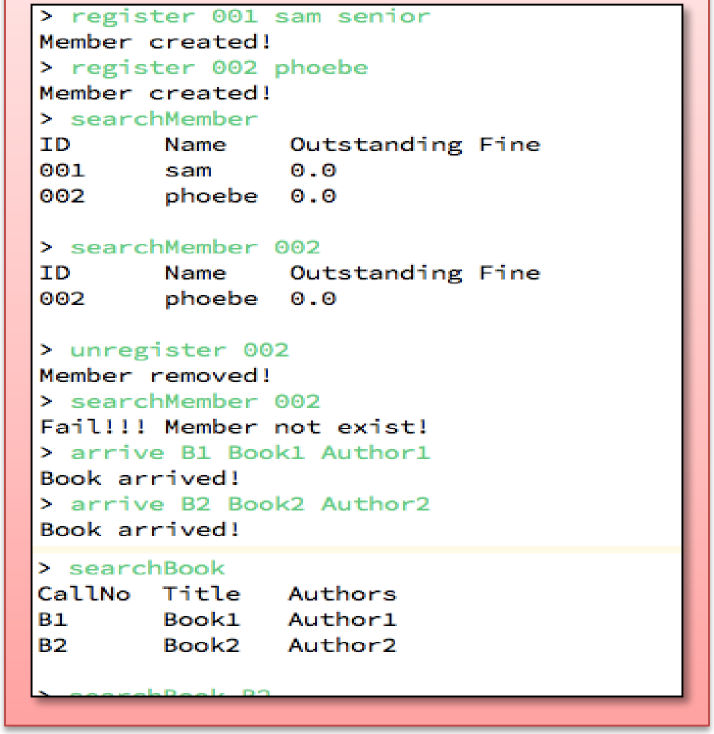
Sample OO Program

Consider a *Library System* which allows:

- Register a new member. A member may be a child, adult or senior.
- Cancel, search for an existing member.
- Add a new book.
- Remove the record of a book.
- Search for the details of a book.
- A member borrows / returns a book.
- A member pays fine. Fine rate is $3/day for children, $10/day for adult and $5/day for senior.
- Undo the last action performed by the user.

Procedural approach and **OO approach** are very different!! Which would be our approach for even *larger* problems?

Sample rundown:



```
> register 001 sam senior
Member created!
> register 002 phoebe
Member created!
> searchMember
ID       Name     Outstanding Fine
001      sam      0.0
002      phoebe   0.0

> searchMember 002
ID       Name     Outstanding Fine
002      phoebe   0.0

> unregister 002
Member removed!
> searchMember 002
Fail!!! Member not exist!
> arrive B1 Book1 Author1
Book arrived!
> arrive B2 Book2 Author2
Book arrived!
> searchBook
CallNo   Title    Authors
B1       Book1    Author1
B2       Book2    Author2

> searchBook B2
```

CS2310 [**Procedural approach**]: Specify **what** tasks to do in each step
CS2312 [**Object-oriented approach**]: Specify **who** performs **what tasks** in each step.

> "Object-oriented design has been widely adopted by businesses around the world. When done properly, the approach leads to simpler, concrete, robust, flexible and modular software. " -- Robert C. Martin *(Uncle Bob)*