Course Syllabus

Course Number	210215
Credits	3 (2-3-6) Cr
Course Title	Programming Methodology
Faculty	Department of Computer Engineering, Faculty of Engineering
Semester/Year	2nd /2021
Instructors	Section 1: Vishnu Kotrajaras (VKJ); Section 2: Chate Patanothai (CNP); Section 33: Peerapon Vateekul (PVK);
Conditions	-
Status	Required
Curriculum	B.Eng.
Degree	Bachelor
Hours/Week	Wed, 13.00-16.00 (3 hours)
Course Description	This course aims at developing individual advanced programming skills. Students are required to have a basic programming background, such as data types, conditional and iterative control flows, creating and using subroutines (methods), and arrays. Important concepts focus in this course including object-oriented design, decomposition, encapsulation, abstraction, exception, thread, event-driven programming, and testing. Students will learn all the concepts through Java programming language along with good software engineering principles, such as Testing Driven Development (TDD) via JUnit-Test-Case. Emphasis is on good programming style and the built-in facilities of the Java language.
Learning/Behavioral Objectives	Students should: • Understand classes and objects. • Be able to use class methods and data from existing classes. • Be able to use Object-Oriented concepts including inheritance, polymorphism, and interface. • Be able to prevent unexpected errors by correctly using Java exception: try-catch and throws. • Be able to develop a responsive Graphical User Interface (GUI). • Be able to use JUnit-Test-Case.

Learning Contents

#	Wed	Title	Topic	Note
1	12 Jan	Lecture 0	Basic Java	
2	19 Jan	Lecture 1	OOP + Exception + Exercise 1 Note: Please download the following software/libraries before coming to class: - JDK 17.0.1 - Eclipse 2021-12 - JavaFX 11 to 15 - Violet UML Editor	UML introduced, Exception usage (just use)
3	26 Jan	Lab1	Eclipse, GitHub, Exception, etc	View GitHub Tutorial
4	2 Feb	Lecture 2	Inheritance + JUnit+ Exercise 2	
5	9 Feb	Lab2	Inheritance + JUnit + Exception	Debug, IO (file)
6	16 Feb		Holiday	
7	23 Feb	Lecture 3	Abstract class + Exercise 3	
7	2 March	Lab3	Abstract class + Writing JUnit.	
8	11 March		Midterm Exam : 08.30-11.30	
9	16 March	Lecture 4	Interface (Polymorphism) + Exercise 4	
10	23 March	Lab4	Interface lab + Exception	
11	30 March	Lecture 5 + Lab 5	GUI (Form; Fx) + Exercise 5	
12	6 April		Holiday	
13	13 April		Holiday	
14	20 April	Lecture 6	Event Handling + Thread + Exercise 6	
16	27 April	Lab6	Event Handling + Thread	
17	4 May		Holiday (Lecture 7 self-study)	
18	12 May		Final Exam : 13.00-16.00	
19	To-be-Announced		Project Presentation	

Teaching Methods Lecture, in-class practice, Lab session, a meeting with Personal Programming Tutor (PPT) (on Discord) every week.

Media On-screen display of presentation slides and programming demonstration. Lecture videos are also available.

Assignments Assignments might be assigned by the instructor of each section.

LMS CourseVille (http://www.myCourseVille.com)
FB Group: "https://www.facebook.com/groups/1003627683818973"
Discord: https://discord.gg/N8JAJ6hDHN

Evaluation **Assessment of academic knowledge:**

- Lab Assignments 30 %
- Exercise participation (PPT session) 5 %
- Class attendance 10 %
- Project 25 %
- Midterm 15 %
- Final 15 %

Scoring criteria In the scoring of each item used for student assessments, instructors will evaluate students' understanding based on students' program behavior, written answers, considering related learning/behavioral objectives as well as correctness of the submitted works.

Grading Letter grades will be assigned based on the total score percentage of each student according to the following table.

Score percentage range (From 100%)	*Letter grade*
[85,100]	A
[80,85)	B+
[75,80)	B
[70,75)	C+
[65,70)	C
[60,65)	D+
[50,60)	D
[0,50)	F

Required Textbook: -

Attendance Students with their attendance below 80% are prohibited from attending the final examination unless the instructors permit.