

Object-Oriented Design

Special Subjects
Elective 2 credit

MIZUTANI, Kozo

1. Course Description

In this course, we learn design patterns for software development by object oriented programming. Not only object oriented programming, but also software development requires design and implementation that achieve re-usability, maintainability, and portability at high level. Applying software design patterns is one of a method to satisfy these requirements. To apply effectively for software development, we need to understand design patterns. The purpose of this course is to understanding GoF design patterns through implementing simple java programs. And also, we aim to become able to choose design patterns adequately, design and implement software for system requirements.

2. Course Objectives

Learners will be able to:

- explain what is software design patterns, understand characteristic of each pattern.
- understand the role and behavior of classes on each design pattern.
- apply design patterns adequately to develop programs.

3. Grading Policy

This course evaluates at a ratio: reports (2 Times) are 50% and an examination is 50%.

4. Textbook and Reference

Textbook

結城浩 Java言語で学ぶデザインパターン入門
SB Creative, ISBN-10:4797316462

5. Requirements(Assignments)

Learners have to learn using prescribed textbooks, and work on questions at the end of each chapter in the textbook for understanding deeply. Although the textbook uses Java, you can use other object oriented programming language. (However, an examination consists questions by Java.)

6. Note

It is needed knowledge of the basis of object oriented programming.

7. Schedule

- | | |
|------|---|
| [1] | Reviewing Object Oriented Programming (Java Programming, UML) |
| [2] | Iterator and Adapter Pattern |
| [3] | Template Method and Factory Method Pattern |
| [4] | Singleton and Prototype Pattern |
| [5] | Builder and Abstract Factory Pattern |
| [6] | Bridge and Strategy Pattern |
| [7] | Composite and Decorator Pattern |
| [8] | Visitor and Chain of Responsibility Pattern |
| [9] | Facade and Mediator Pattern |
| [10] | Observer and Memento Pattern |
| [11] | State Pattern |
| [12] | Flyweight and Proxy Pattern |
| [13] | Command Pattern |
| [14] | Interpreter Pattern |
| [15] | Integrated Study |