PENDAHULUAN

Pengembangan Sistem Berorientasi Object (PSBO)

Agenda

- Penjelasan Kontrak Perkuliahan
- Review Konsep Object Oriented
- Grand Design OOAD

Tujuan Perkuliahan

Mahasiswa mampu mengimplementasikan metode pengembangan berorientasi objek terkini dan mampu membangun aplikasi berorientasi objek dengan menerapkan teknik lanjut dari perancangan dan pemrograman berorientasi objek

Deskripsi Mata Kuliah

1. Prinsip dan konsep dasar orientasi objek seperti ADT, enkapsulasi, *inheritance, information hiding, polymorphism* .
2. Teknik pengembangan perangkat lunak berorientasi objek: analisis orientasi objek dan disain orientasi objek dengan bahasa pemodelan UML

Deskripsi Mata Kuliah

1. implementasi/ pemrograman berorientasi objek dengan Java, penggunaan ulang (r*euse*) software, Perancangan untuk penggunaan ulang, pola perancangan (*design pattern*), *object presistency*
2. studi kasus

Kemampuan Dasar Pemrograman

- Pernah Membuat program dengan Bahasa C/C++ & Java
- Memahami Konstruksi dasar Bahasa pemrograman:
  - Deklarasi Variabel
  - Looping
  - Assignment
  - Procedure & Function

Kemampuan RPL

- Memahami konsep pengembangan PL.
- Telah menerapkan metode PL
- Menguasai notasi DFD, ERD dengan Baik.
- Mengenal dasar-dasar OO.

GBPP

- GBPP Kuliah
- GBPP Praktikum

Aturan Penilaian

- 35% UTS
- 35% UAS
- 30% Praktikum & Tugas Akhir

Bahan dan Sumber

- Bennet, S., McRobb, S., and Farmer, R. 2002. Object-Oriented Systems Analysis and Design Using UML. 2nd edition. McGraw-Hill Education
- Braude, Eric J. 2000. Software Engineering : an Object Oriented Perspective. John Wiley & Sons.
- Booch, G., Rumbaugh J., Jacobson, I. 1999. The Unifield Modelling Language User Guide. Addison-Wesley.
- Bahrami, Ali, 1999. Object Oriented System Development. Irwin McGraw-Hill.

Bahan dan Sumber

- Gamma, E. at all. 1995. Design Patterns: Elements of Reusable Object Oriented Software. Addison-Wesley
- P.J. Deitel, H.M Deitel, 2004,Java: Howto Program 3rd.Prentice-Hall, New Jersey.
- Craig L. 2004, Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design, Addison-Wesley.
- B.D. McLaughlin, G. Pollica, D. West., 2008,Head First Object-Oriented Analysis & Design, O'Reilly
- Eric, F, Elisabeth, F. 2008, Head First Design Pattern. O'Reilly
- Kathy, S., Bert, B., 2005, Head First Java, O'Reilly.

Kuliah Online

- Semua bahan dan materi akan disediakan pada webpage KulOn Ilkom (http://ilkom.fmipa.ipb.ac.id/kulon)
- Enrollment key:psbo2010
- Tugas dan penyerahannya pada website
- Komunikasi antar pengajar/asisten dan mahasiswa

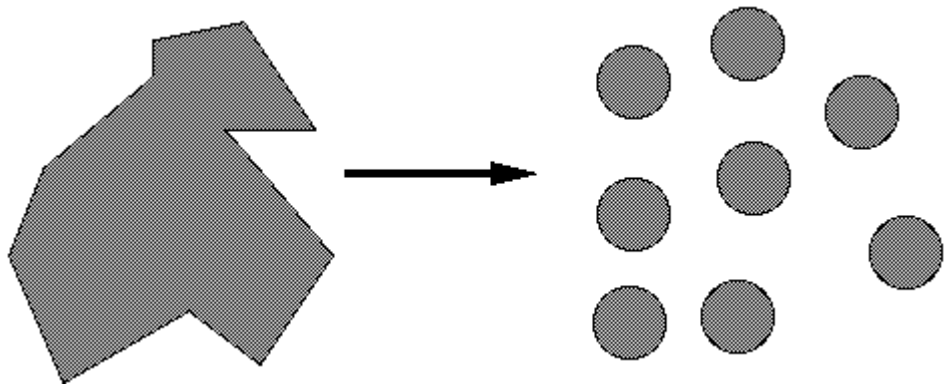
Tugas Akhir/Project

- Pengembangan Sistem software dalam kelompok
- Tema: Imagine Cup / PKM
- Harus menggunakan bahasa pemrograman OOP dan menggunakan konsep-konsep:
  - Analisis dan Desain Berorientasi objek (UML)
  - Class, Inheritance, Abstract class/virtual, polymorphism
  - Pola design (Design Patterns)
  - Java, C++,C#, PHP (dengan OOP), etc.
- Akan disinkronkan dengan MK IMK dan SI
  - More on this later…..

Setuju?

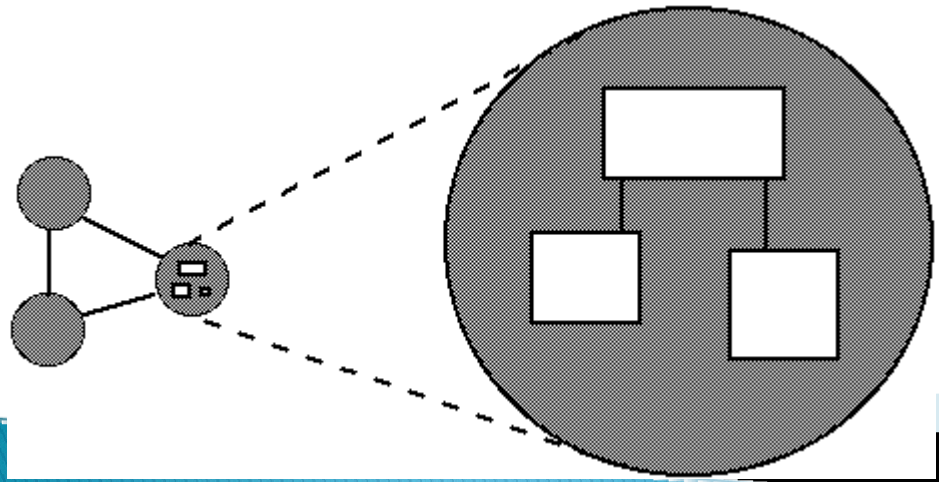
Why OOP

- Modularization: Decompose problem into smaller subproblems that can be solved separately.



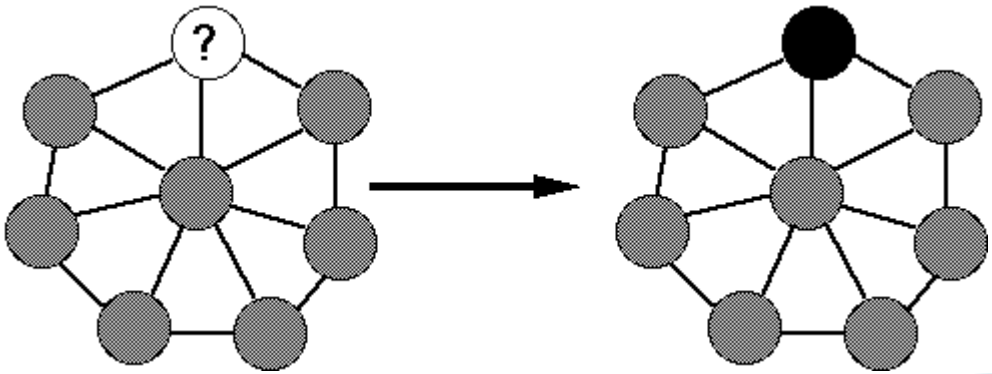
Why OOP?

- Abstraction/Understandability: Terminology of the problem domain is reflected in the software solution.
  Individual modules are understandable by human readers.



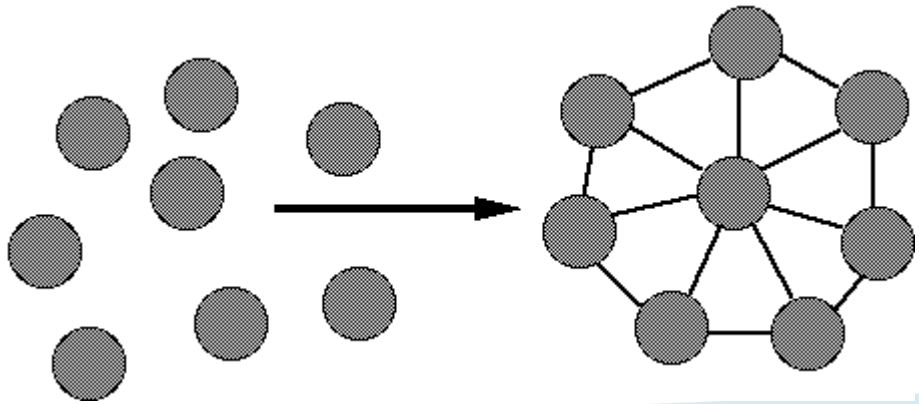
Why OOP?

- **Encapsulation -- Information Hiding**
  - Hide complexity from the user of a software of SDK. Protect low-level functionality.



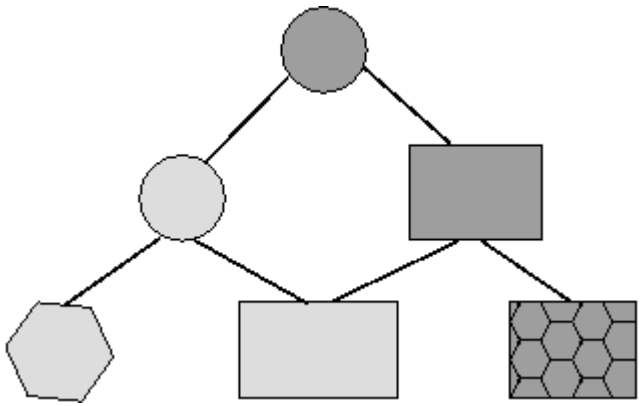
Why OOP?

- Composability
  - Interfaces allow to freely combine modules to produce new systems.



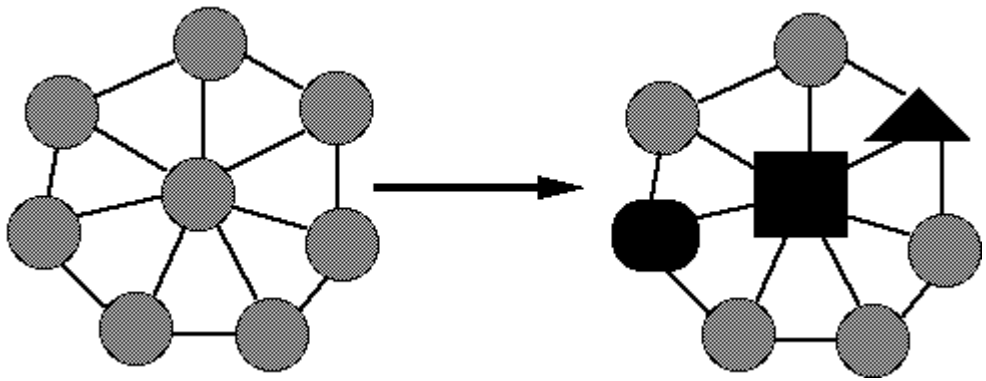
Why OOP?

- Hierarchy
  - Incremental development from small and simple to more complex modules.



Why OOP?

- Continuity
  - Changes and maintenance in only a few modules does not affect the architecture.



Review OOP

- Class – Object
- Attribute – Method
- Concept Encapsulation/Information Hiding
- Inheritance
- Polymorphism

Berikutnya…

- Pengertian lebih mendalam mengenai konsep-konsep dasar PSBO
- Dasar-dasar UML Modelling