

Class Diagram

Bahan Pelengkap sesi 8

Class Diagram

- ▶ www.teknikindustri.org
- ▶ www.agilemethod.com
- ▶ www.parlezuml.com
- ▶ Whitten, ch. 11
- ▶ UML
- ▶ Class Diagram

Unified Modeling Language (UML)

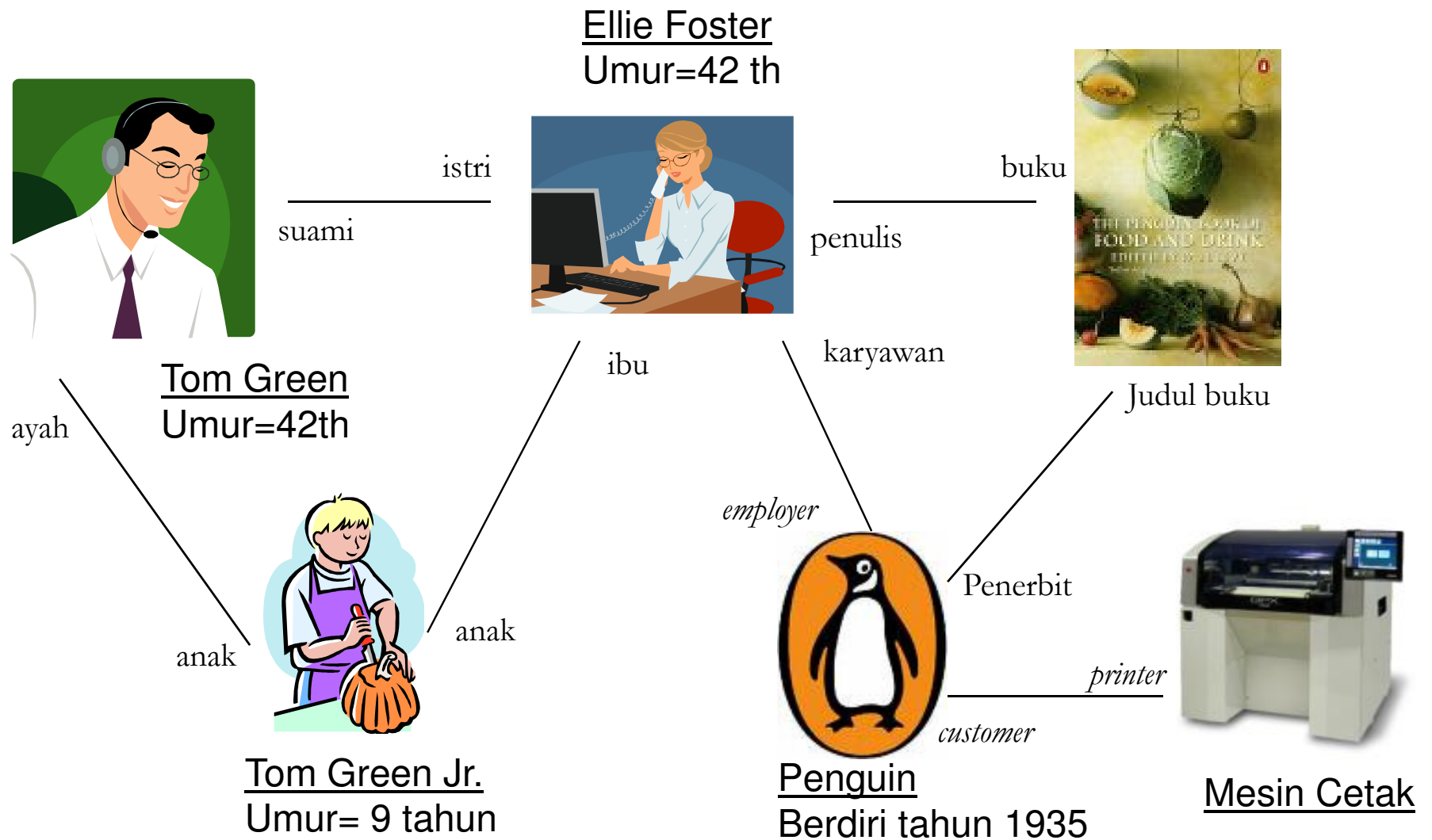
- ▶ UML digunakan untuk menggambarkan obyek, atributnya (fakta yang perlu kita ketahui tentang obyek tersebut), dan relasinya dengan obyek lain pada saat tertentu

UML

Istilah-istilah menyangkut obyek:

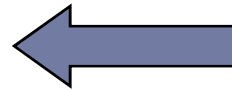
- ▶ **Unique identity:** sebuah obyek memiliki identitas unik yang tidak berubah
- ▶ **Roles:** sebuah obyek memiliki berbagai peran dalam hubungannya satu dengan yang lain
- ▶ **Type/class:** UML dapat digunakan untuk memodelkan tipe atau kelas dari obyek. Tipe adalah sekumpulan obyek yang memiliki karakteristik sama.
- ▶ Kita dapat memodelkan **constraint** rumit yang diaplikasikan dalam satu tipe dengan cara menambahkan **rule** (aturan) dalam tipe tsb

UML



UML Diagram

- ▶ **Use-Case Model Diagrams**
- ▶ **Static Structure Diagrams**
 - ▶ *Class diagrams*
 - ▶ *Object diagrams*
- ▶ **Interaction Diagrams**
 - ▶ *Sequence diagrams*
 - ▶ *Collaboration diagrams*
- ▶ **State Diagrams**
 - ▶ *Statechart diagrams*
 - ▶ *Activity diagrams*
- ▶ **Implementation Diagrams**
 - ▶ *Component diagrams*
 - ▶ *Deployment diagrams*

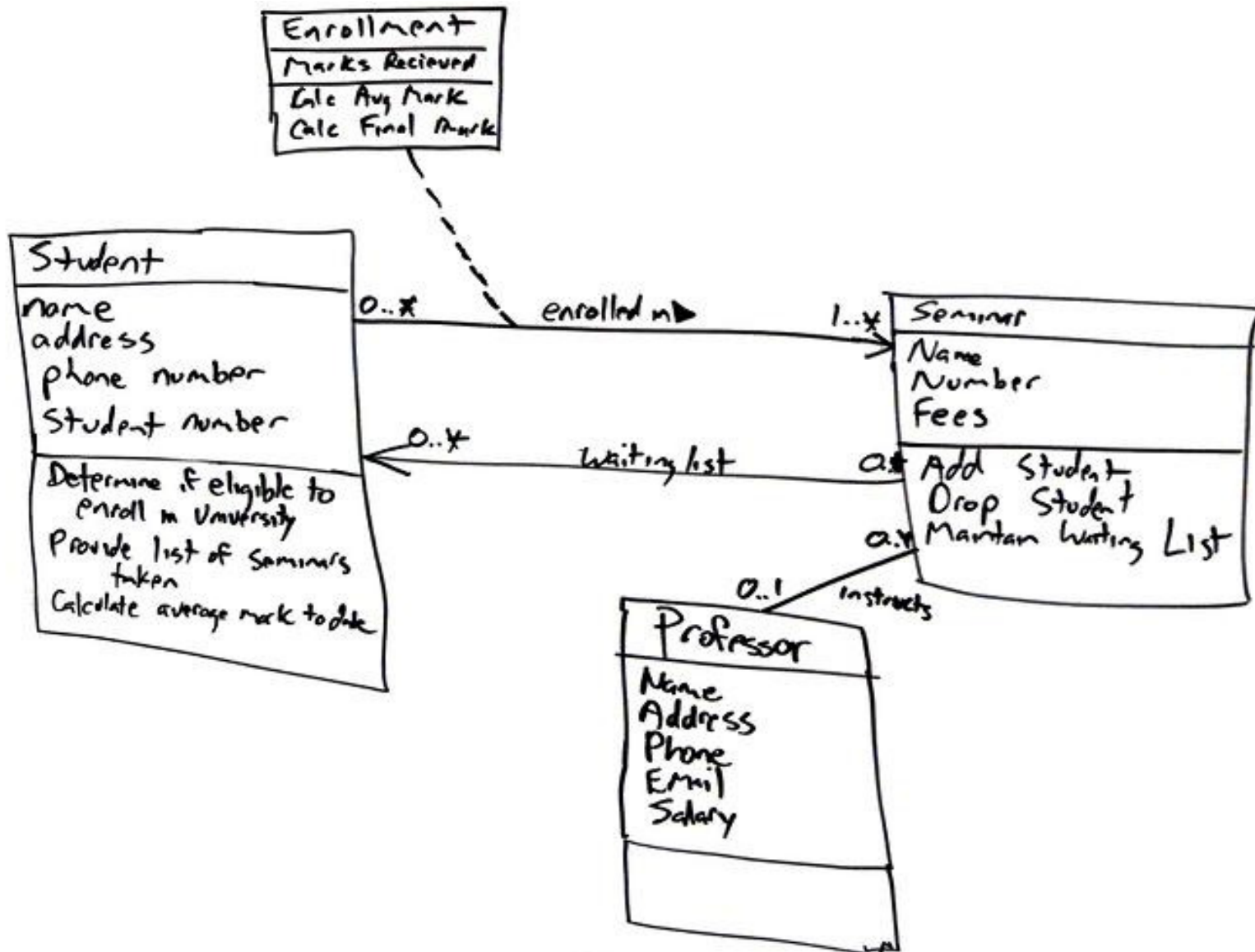


Class Diagram

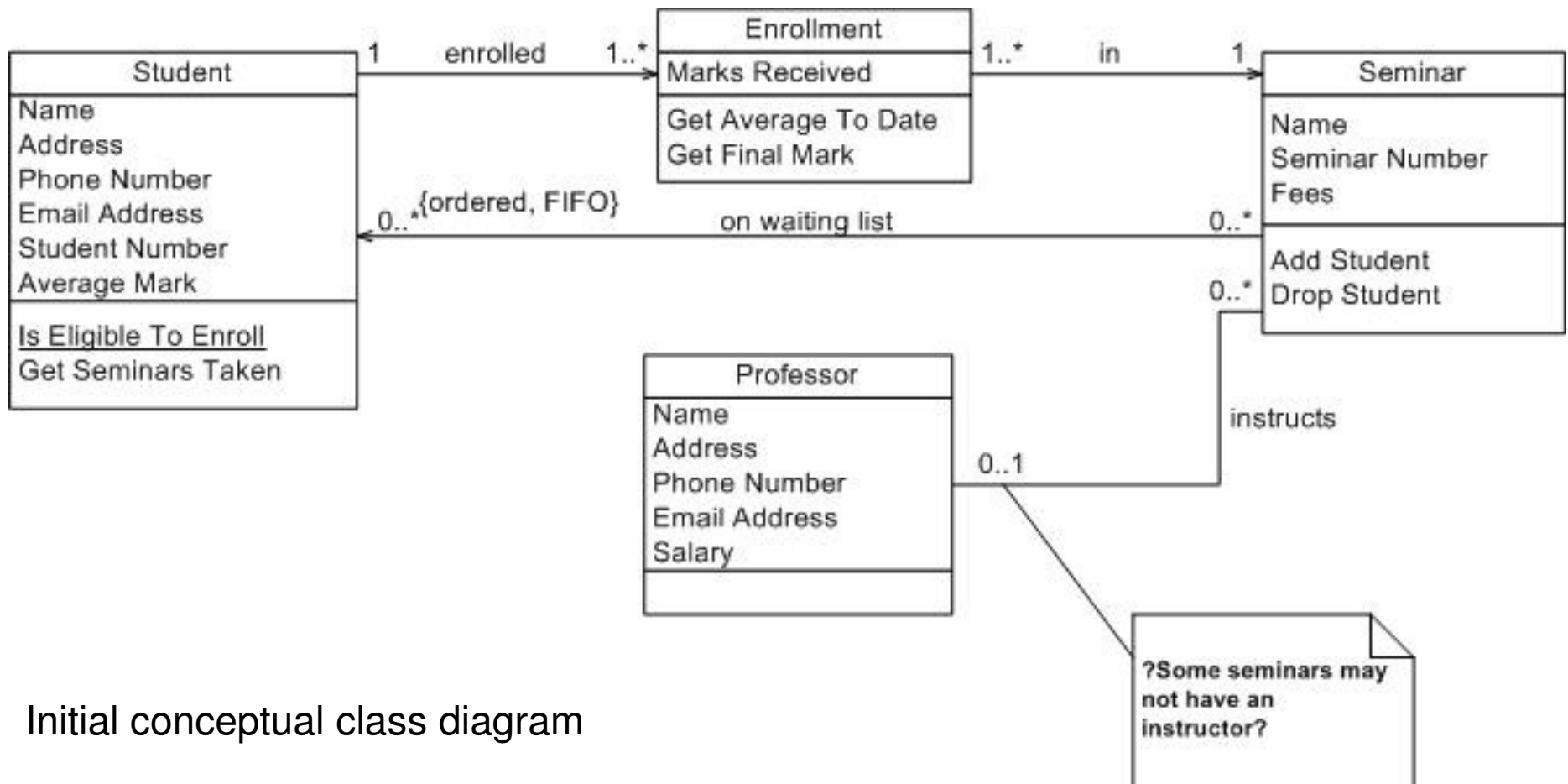
- ▶ Sebuah diagram kelas menggambarkan obyek dalam sebuah sistem, dimana terdapat relasi antar tipe/kelas.
- ▶ Sebuah obyek harus mematuhi aturan yang berlaku dalam kelas/tipe tersebut

Class Diagrams

- ▶ **Association:** menggambarkan hubungan antar obyek dari tipe yang berbeda.
- ▶ **Generalisation:** menggambarkan bahwa satu kelas merupakan bagian dari kelas yang lain (*subclass/subtype*)



Sketch of a conceptual class diagram



Initial conceptual class diagram

How to create class diagram

- ▶ **Classes**
- ▶ **Responsibilities**
- ▶ **Associations**
- ▶ **Inheritance relationships**
- ▶ **Composition associations**
- ▶ **vocabularies**

Class Diagram - Classes

- ▶ Obyek: orang, tempat, benda, konsep, kejadian, layar/screen, laporan/report dalam satu sistem
- ▶ **Obyek** memiliki **atribut** dan **metoda**

Class Diagram - Responsibilities

- ▶ Sebuah kelas digambarkan dengan kotak yang berisi tiga bagian:
 - ▶ Nama kelas
 - ▶ Atribut
 - ▶ Metoda
- ▶ Contoh: mahasiswa (**kelas**) memiliki NIM, alamat, dan nomor telepon (**atribut**). Mahasiswa mendaftar kelas, membatalkan kelas, meminta transkripsi nilai (**metode**)

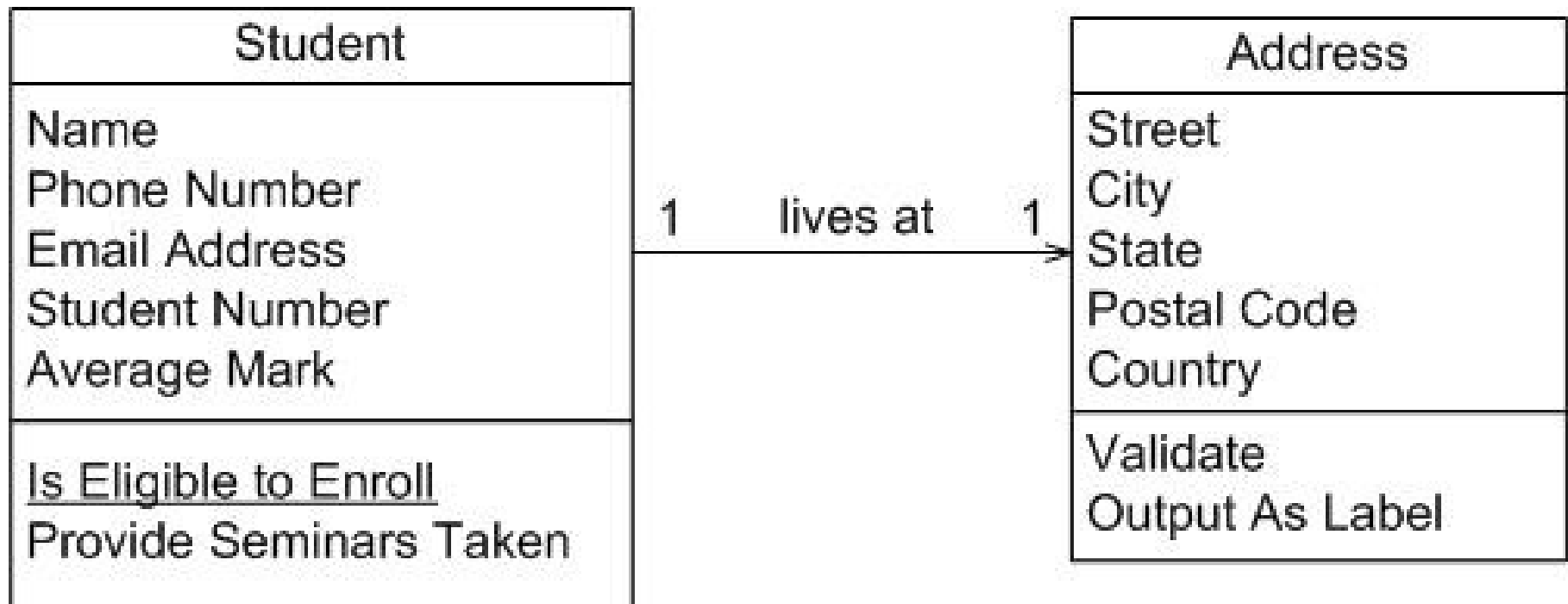
Class Diagram – Responsibilities (cont)

- ▶ Mahasiswa/student memiliki alamat. Alamat memiliki informasi kompleks: nama jalan, kota, propinsi, kode pos.
- ▶ Agar kelas Mahasiswa menjadi lebih kohesif, tanpa memiliki atribut alamat yang kompleks, maka kita memodelkan alamat menjadi kelas tersendiri, dengan atribut: nama jalan, nomor, kota, propinsi, kode pos

Class Diagram – Responsibilities (cont)

- ▶ Kohesivitas kelas mahasiswa dapat terjaga, karena kita melakukan metode validasi (memeriksa logika kebenaran alamat) dan mencetak alamat dimasukkan dalam kelas alamat.
- ▶ Cara yang sama kita lakukan untuk membuat relasi antara kelas seminar dengan kelas kuliah/course

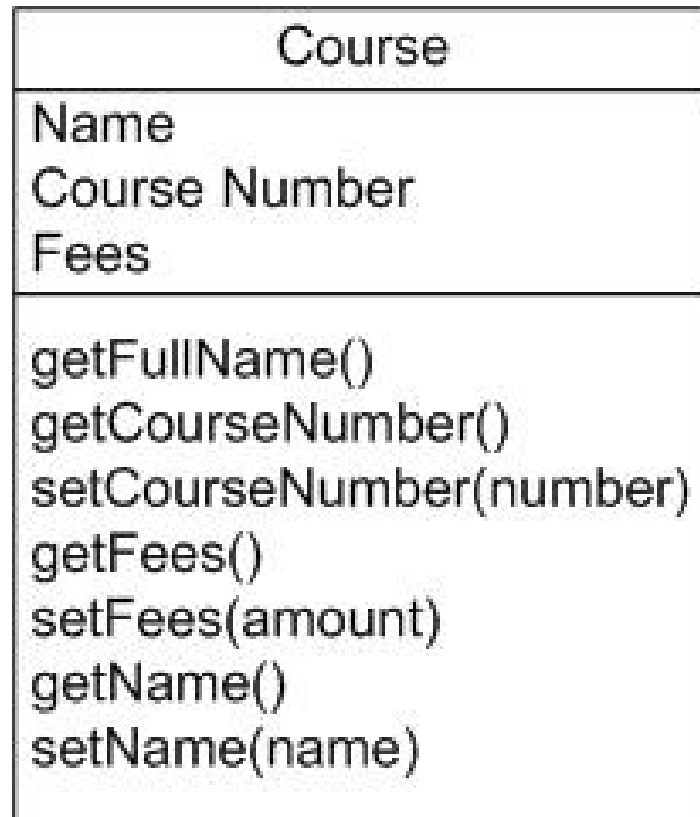
Class Diagram – Responsibilities (cont)



Class Diagram – Responsibilities (cont)



Class Diagram – Responsibilities (cont)



Class Diagram - Association

- ▶ Obyek berhubungan dengan obyek lain
- ▶ Hubungan digambarkan dalam garis yang menghubungkan antara dua kelas
- ▶ Label (meski opsional) biasanya dalam satu atau dua kata menggambarkan asosiasi (relasi)

Class Diagram – Association

(cont)



Class Diagram – Association

(cont)

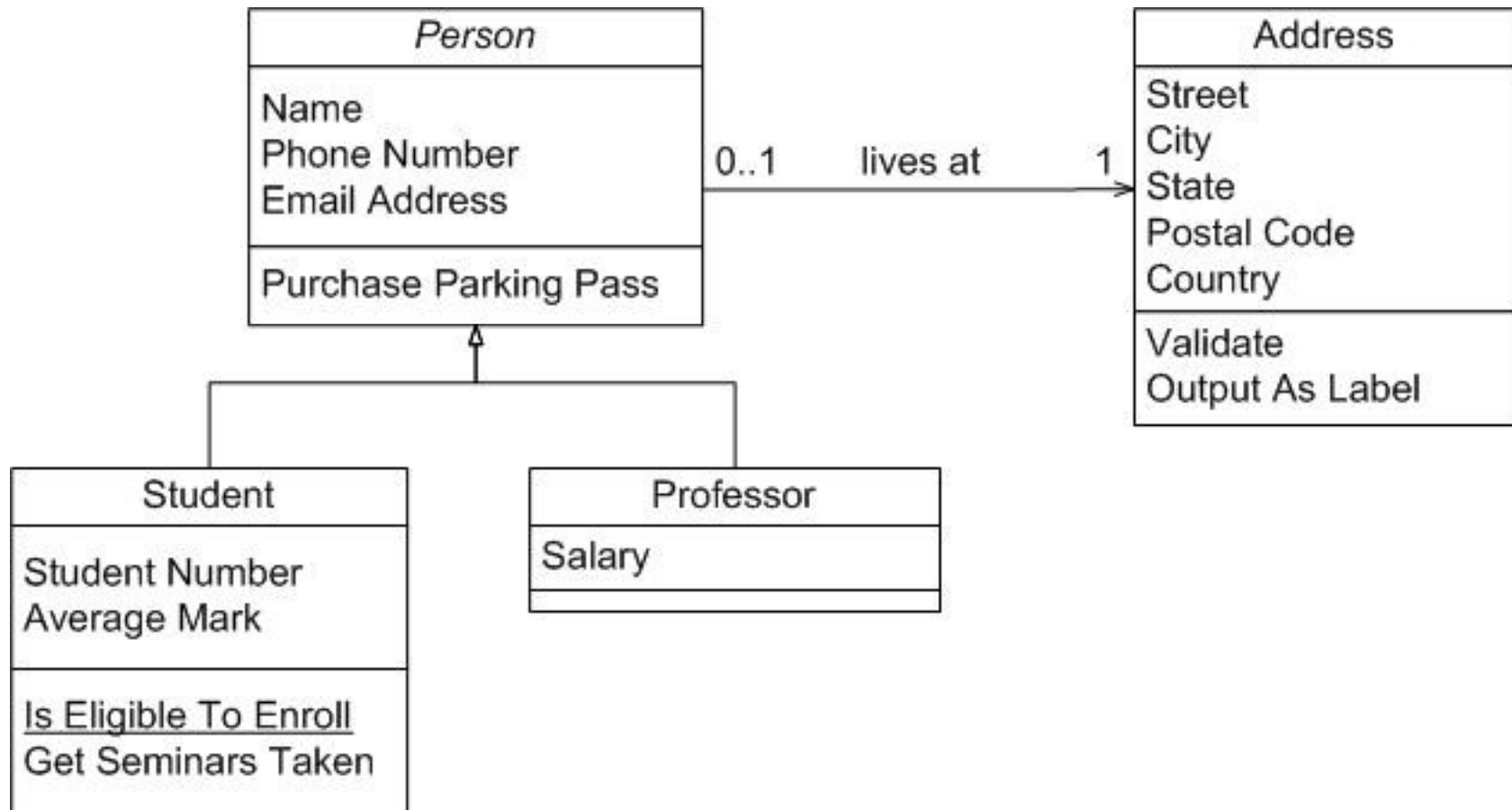
Table 1. Multiplicity Indicators.

Indicator	Meaning
0..1	Zero or one
1	One only
0..*	Zero or more
1..*	One or more
n	Only n (where $n > 1$)
0.. n	Zero to n (where $n > 1$)
1.. n	One to n (where $n > 1$)

Class Diagram - Inheritance

- ▶ Seringkali satu atau lebih kelas memiliki atribut dan/atau metode yang sama.
- ▶ Kita tidak perlu menuliskan kode yang sama berulang kali, sehingga digunakan mekanisme **inheritance**
- ▶ *If A inherits from B \rightarrow A is the subclass of B, B is the superclass of A*
- ▶ **Pure inheritance**: jika A mewarisi seluruh atribut dan metode dari B. Contoh: mahasiswa dan dosen

Class Diagram – Inheritance (cont)



Class Diagram – Composition Associations

- ▶ Kadang sebuah obyek terdiri dari obyek yang lain
- ▶ Contoh: misalkan sebuah gedung/building terdiri dari satu atau beberapa ruangan, dan ruangan terdiri dari beberapa sub-ruangan
- ▶ Ruangan adalah bagian dari (part of) gedung. Tapi sebuah alamat BUKAN merupakan bagian dari mahasiswa.

Class Diagram – Composition Associations (cont)



Class Diagram - Vocabularies

- ▶ Sebuah *vocabulary* menjelaskan
 - ▶ semantik dari tipe entitas dan *responsibility*-nya
 - ▶ Relasi taksonomikal antar tipe entitas
 - ▶ Relasi ontologikal antar tipe entitas
- ▶ Taksonomi: klasifikasi entitas dalam hirarki. Contoh: taksonomi entitas *person*
- ▶ Ontologi di atas taksonomi menggambarkan pengetahuan tentang topik dan relasi yang dimiliki entitas dalam topik yang dimaksudkan.

Class Diagram – Vocabularies (cont)

