

OOP with Java

Hsuan-Tien Lin

Department of CSIE, NTU

OOP Class, April 28, 2009

Can we BE more OOP?

Sure, but what do you want to learn?

How to use Java to **realize** OOP?

- abstract OOP thinking: UML
- concrete OOP realization: Java

How to use Java to **realize** OOP & to **write** programs?

- a modern and useful 1st language: C
- a modern and useful 2nd language: Java

How to use Java to **realize** OOP &
to **write** programs?

- the mechanism within a language (and the platform)
- almost your only chance because there is no PL class now

The Missing Part, OOP-Wise

- how does one design good OOP program (emphasized in recent homeworks)?
- **how does the mechanism relate to OOP?**

Let's go with a bigger picture.

abstraction

the ability to express and manipulate objects

- class for blueprint (as the “class” file)
- instance for individual objects (in memory)
- reference (a.k.a. type-safe pointer) for the “handle” of object
- object lifecycle: from constructor to finalizer

abstraction

the ability to express and manipulate objects

- data (a.k.a. memory interpretation):
important for program designers
not so important for component users (see encapsulation)
- action (a.k.a. method invocation): pass in `this` for
instance-specific actions

composition

the ability to play and interact with objects

- didn't emphasize it, but naturally just used it
- object accessing scheme (use existing classes through static/instance/local variables)
—again, object lifecycle
- action invocation scheme (stack frame, parameter passing, and return value)
—again, instance-specific (and independent) actions

inheritance

the ability to extend objects

- type compatibility
 - hierarchy in a full-OO language
- same reference, compatible types, how?
 - possibly, shared prefix mechanism
- same reference, overridden actions, how?
 - will talk more in polymorphism

encapsulation

the ability to “pack” objects

- as an advanced abstraction
 - safer (and more clear) to be “public” only when necessary
- language/platform support for different access levels