Multiple inheritance

Sometimes single inheritance isn't convenient

- a class might be "a kind of" more than one other class
- might want to reuse the impl of more than one other class

Examples:

`Square` is a kind of `Rectangle` and a kind of `Rhombus`

`ReadWriteStream` is a kind of `ReadStream` and a kind of `WriteStream`

`Array` is a kind of `KeyedCollection` and a kind of `OrderedCollection`

`TitledBorderedWindow` is a kind of `Window` and wants to "mix in" the code for `TitledWindow` and `BorderedWindow`

Language support for multiple inheritance

Allow a class to have a list of superclasses

Methods of the subclass are the union of the methods of the superclasses, extended and/or overridden by subclass methods

Instance variables of the subclass are the union of the inst. vars of the superclasses, extended by subclass inst. vars

Problem with multiple inheritance: ambiguity

What if more than one superclass defines a method with a particular name?

What if more than one superclass declares an inst var with a particular name?

Examples:

- both `Rectangle` and `Rhombus` define `print` methods
- both `Rectangle` and `Rhombus` define `center` inst vars

Typical "solutions" for duplicate methods

Ordered multiple inheritance (Common Lisp)

- order the superclasses
- search superclasses in order, take first match

Unordered multiple inheritance (Extended Smalltalk, C++)

- report an error if ambiguous
- special case:
  if methods all inherited from same original method (diamond-shaped hierarchy), don't mark it ambiguous
- provide other language features to explicitly resolve ambiguities (e.g., special kinds of `super` sends)
- more work for programmer,
  but more error detection by the machine

Unordered multiple inheritance of interfaces (Java)

- interfaces: classes with only abstract methods
- merge "duplicate" methods,
  since no implementation to combine

Typical “solutions” for duplicate inst. var decls

Merging (Common Lisp)

- merge duplicates into one shared inst var

Same source only (C++)

- if inst. vars from same original declaration
  (i.e., diamond-shaped hierarchies), then merge
  (technically, only if from virtual base class)
- otherwise, report error

Multiple inheritance of interfaces (Java)

- no instance variables allowed in interfaces