Types in languages: Smalltalk

Amer Diwan

What is Smalltalk

• A dynamically-typed object oriented language
• Everything is an object
  – Data
  – Control structures (if, while, etc.)
  – Code blocks
  – Classes
• Garbage collection
Example hierarchy

```
Object
  \rightarrow Number
    \rightarrow Integer
    \rightarrow Small
    \rightarrow Bigint

  \rightarrow Collection
    \rightarrow Real
```

Run-time “type compatibility”

- Defined by the set of methods that may be invoked
- Consider invoking method m of object o
  - If o has an m method, then it is invoked
  - Otherwise a “method not understood” message
Metaclasses: the class of a class

• Useful for:
  – Allocation and initialization of new objects of a class
  – Keep track of all objects of a class
  – ...

An example

[Diagram showing class relationships and examples]
Discussion

• How does this compare to C++/Java/Modula-3 model?
  – Power?
  – Conciseness?
  – Speed?

Implementation implications

• Everything is an object
  – Potentially slow
  – (more on this later)
Next lecture: Polymorphism

- Polymorphism:
  - Kinds of polymorphism
- Reading: Cardelli and Wegner (follow link on syllabus page)
  - A dense paper: give yourself some time!