CS 334
lecture 14
OOP Recap
Object = Data + methods

4 Features
-dynamic dispatch
-encapsulation
-subtyping. A <: B if A responds to all messages B responds to.
-inheritance.
Small Talk!
Xerox PARC, 1972-1980

Design Principles

1. Good design.
2. Communicating objects.
3. Uniform Metaphor
5. Extensibility / Polymorphism.
6. User Interface.
   - what is an OS? There shouldn't be one.
   - Reachve.
selectors:

- Infix notation:
  
  \[
  \text{moveDx:}_x \text{ Dy:}_y = \text{moveDx Dy} (\_, \_)
  \]

Other Notation

- \(\leftarrow\) is assignment.
  - (underscore also assignment)

- \(\mid\ldots\mid\) - local vars.
  - \(^\wedge\) is return.
Runtime Representation:

0. Creation:
   - space for fields
      (from template)
   - ptr to metadata for class

1. Methods:
   - lookup in dictionary

2. Field lookup:
   - use template layout.
Subclass Dictionary

1) New methods
2) Redefined methods.

Dynamic Lookup
- Look in own method dictionary
- Look in super method dictionary.
Encapsulation
- methods are **public**.
- fields are **protected**.

Subtyping
Protocol : set of selectors an object responds to

Point obj: \{ x:y, move Dx:Dy, x,y, draw \}
CPoint obj: \{ " " " " " " " " color \}

- expr is free of type errors if all message sent to objects can be received (are in their protocols.)
pt X →
cpt color →
pt color X → message not understood.

Protocol conformance:

A <: B if protocol of B ≤ protocol of A.

CPoint <: Point.

Subtyping not tied to inheritance.
Inheritance
- yes, we have it.
Object Metaphor

-all data are objects.

$X + 2 \Rightarrow X$ plus: 2

-control structures

10 times Repeat: $[a = a + 2]$

"Function" object w/ an eval method.