09/09

How much memory to be allocates depends on the structure of the class.

address of each object defined in class

static time - can make the decision by looking at the source code.

foo(int, boolean) foo (int, int)

call foo (5, true) - can I statically decide which function will be called (yes)

- ~~Com~~ Static Polymorphism / Overloading / Compile Time Polymorphism

A x;
x foo(); can't say which foo is called when the same fn is declared in two class and one is the child of the other.

- Run time / Dynamic Polymorphism (Overriding)

No polymorphism for fields (Java)
Fields / Bounds are resolved statically
Method - decided at run time (resolved dynamically (in Java)
Java has lexical scoping

(define (Animal arg)
  :
  return lambda.
)

to return an inherited class.

(define (Monkey) ...

if I find a ~~fn~~ method that is not overridden I don't go to the parent, now I do.

have a (lambda (msg) in the child.

(cond ((eq? msg 'tail) (tail))
(cond ((eq? msg 'sound) ← call your own tail method.
(super (sound)).
↓ points to an Animal object.

let ((super (Animal

Whenever we create a child object, we create a parent object for overriding cases.