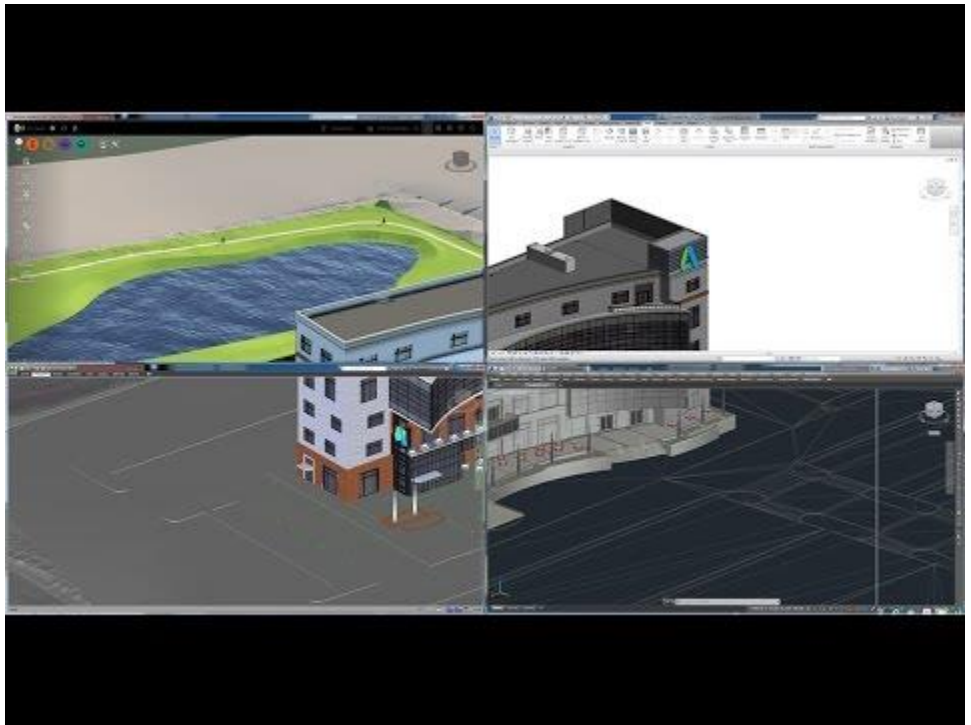
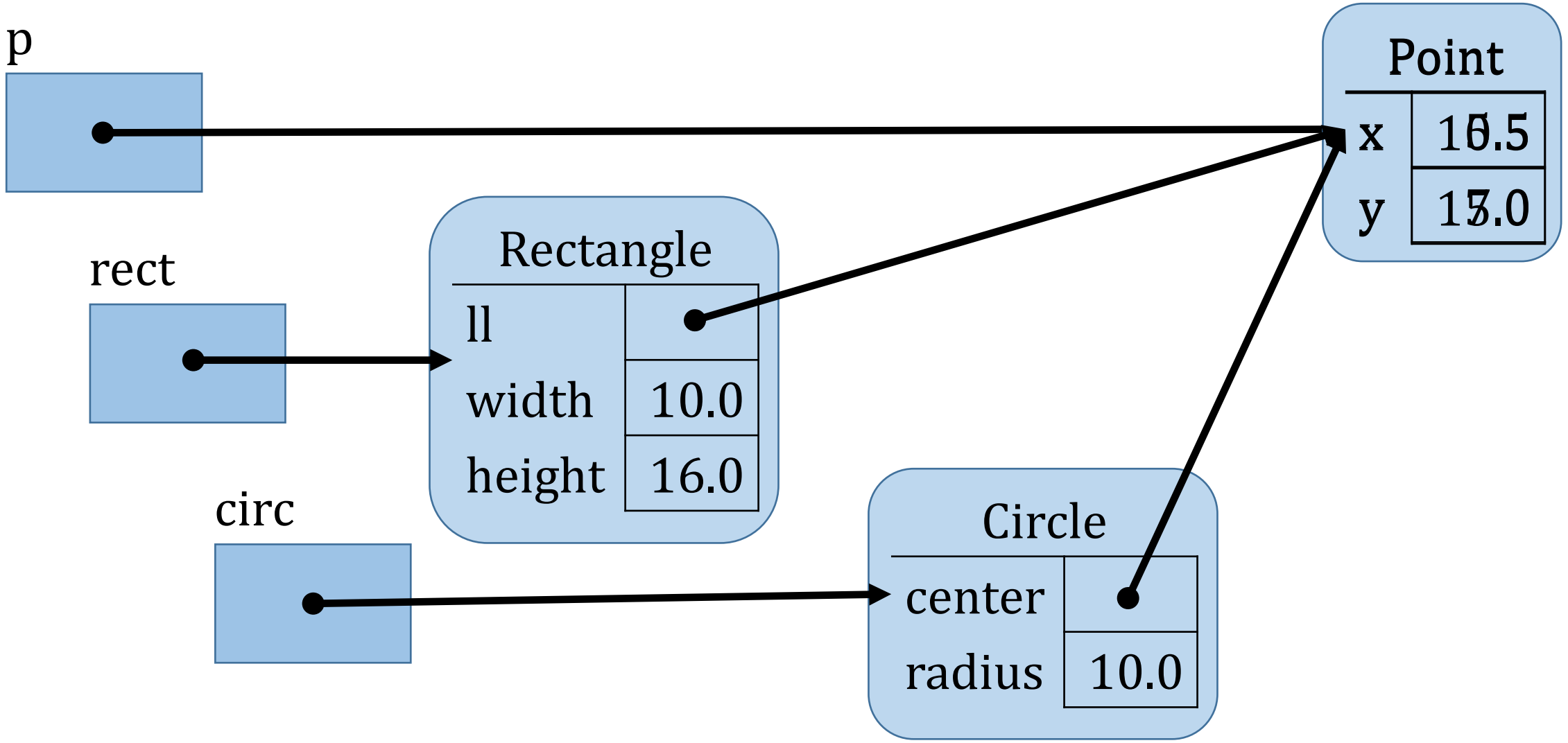
Shared References



Problem

Make a Point in an (x,y) coordinate system that can be moved. Then make a Rectangle with a Point as the lower left corner, a width, and a height; and a Circle with a Point as the center and a radius. Demonstrate by making a TestShapes with a main that creates a Rectangle and a Circle in the center of the Rectangle.

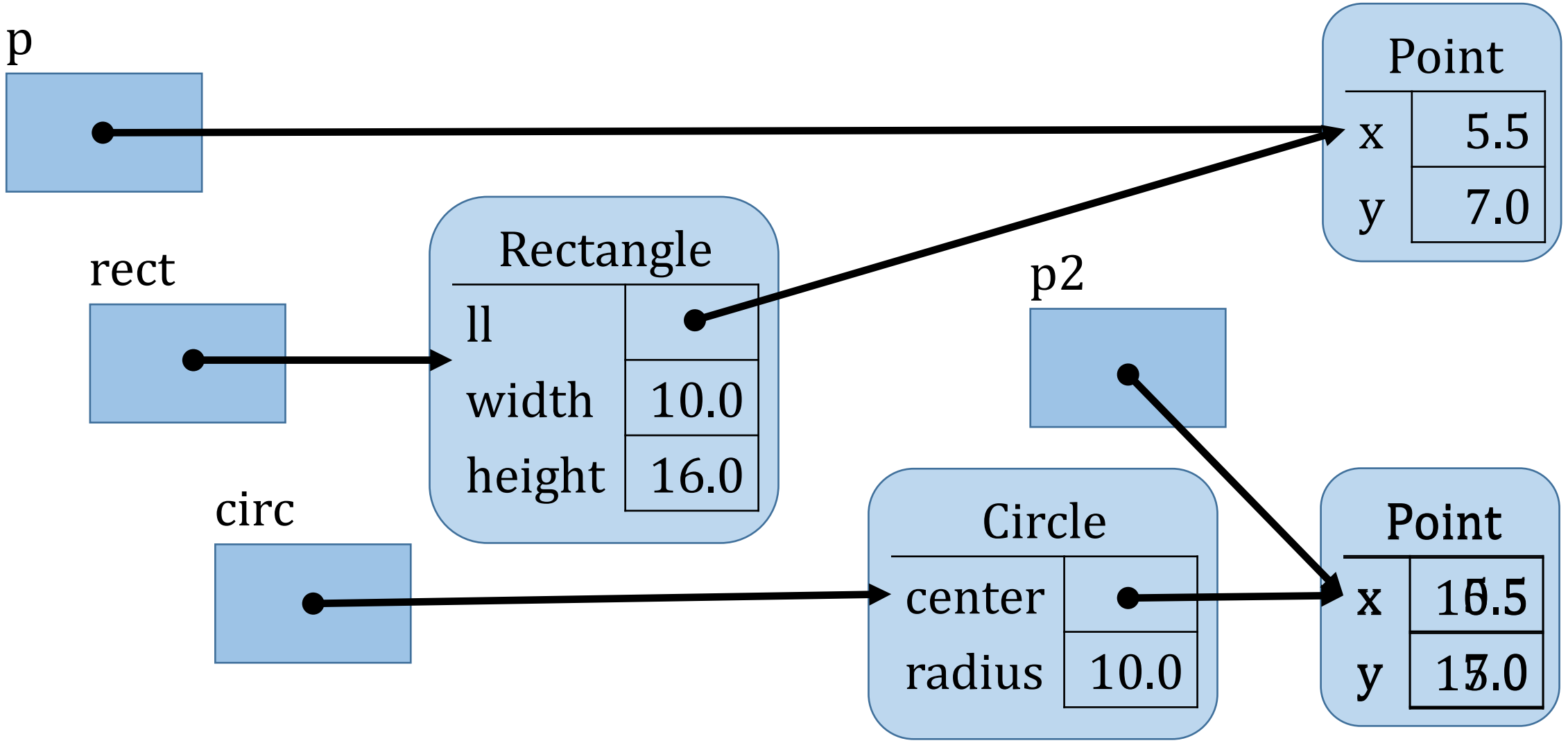
Schematic Shared Reference



Fix 1A: Reference a Copy

- Change "main" to create a second point object for the circle
- Requires very sophisticated user!

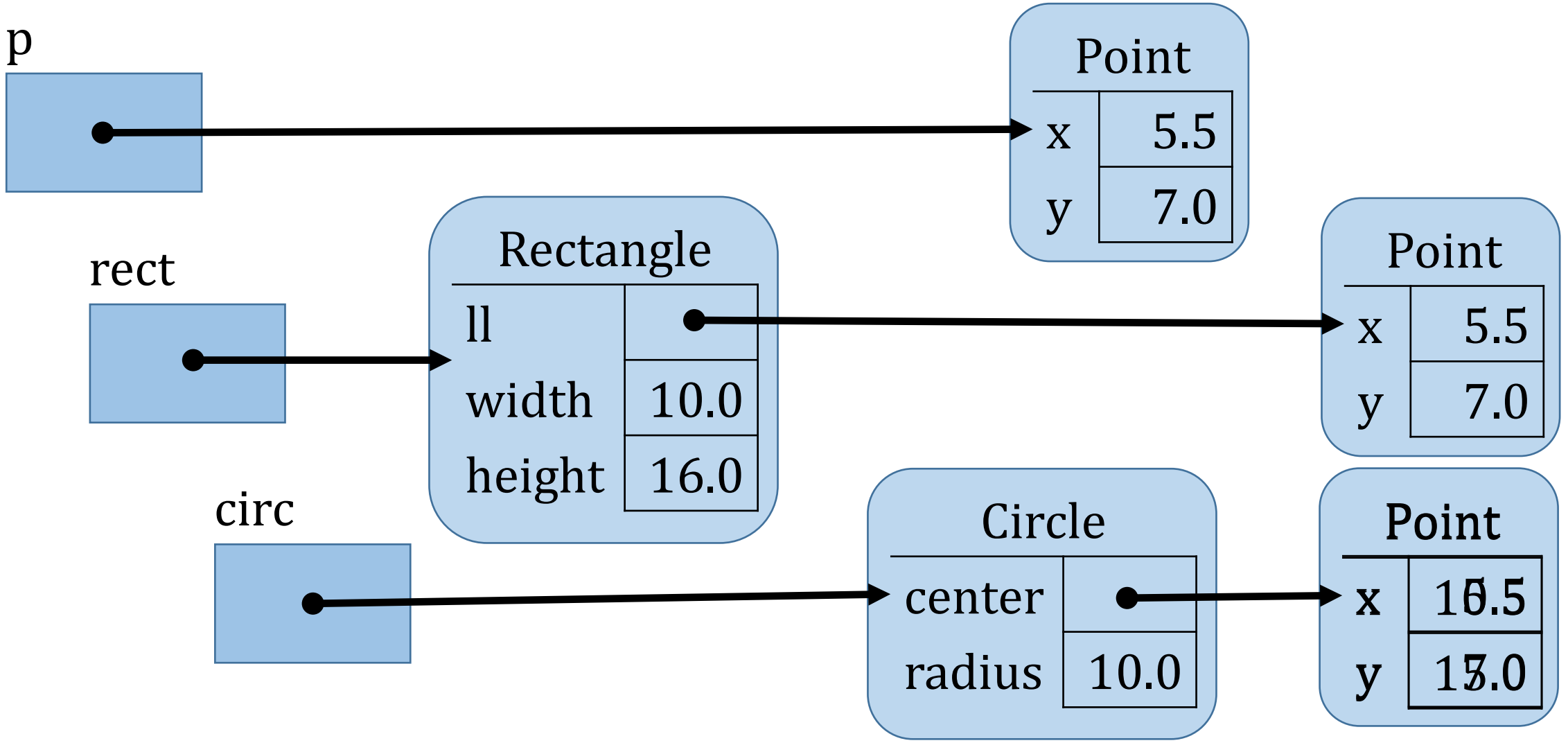
Schematic Shared Reference



Fix 1B: Reference a Copy

- Need a "copy constructor" or "clone constructor" in the point object
- Modify Rectangle and Circle to reference a copy of the Point, not the parameter Point
- May be overkill… duplicates objects

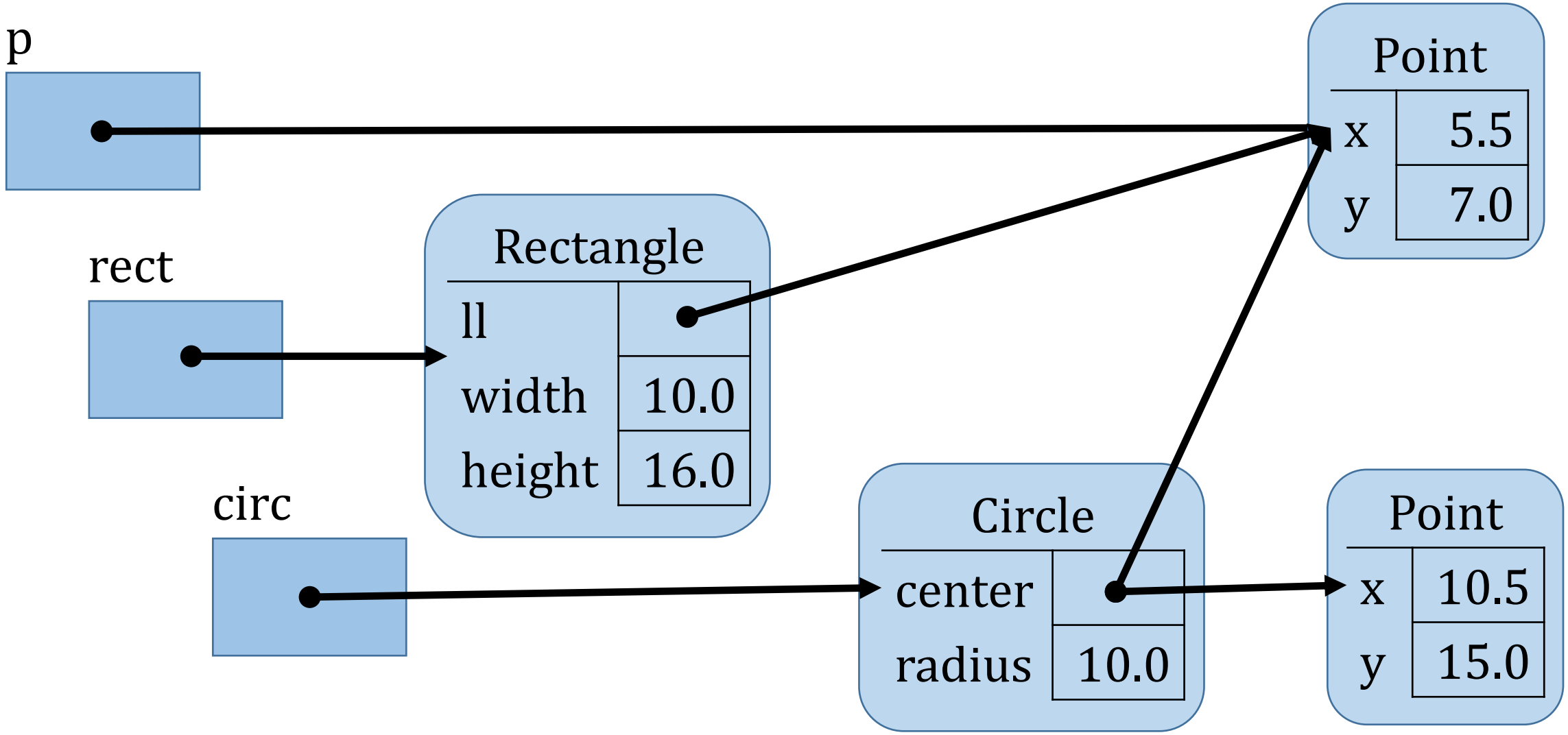
Schematic Shared Reference



Fix 2: Immutable classes

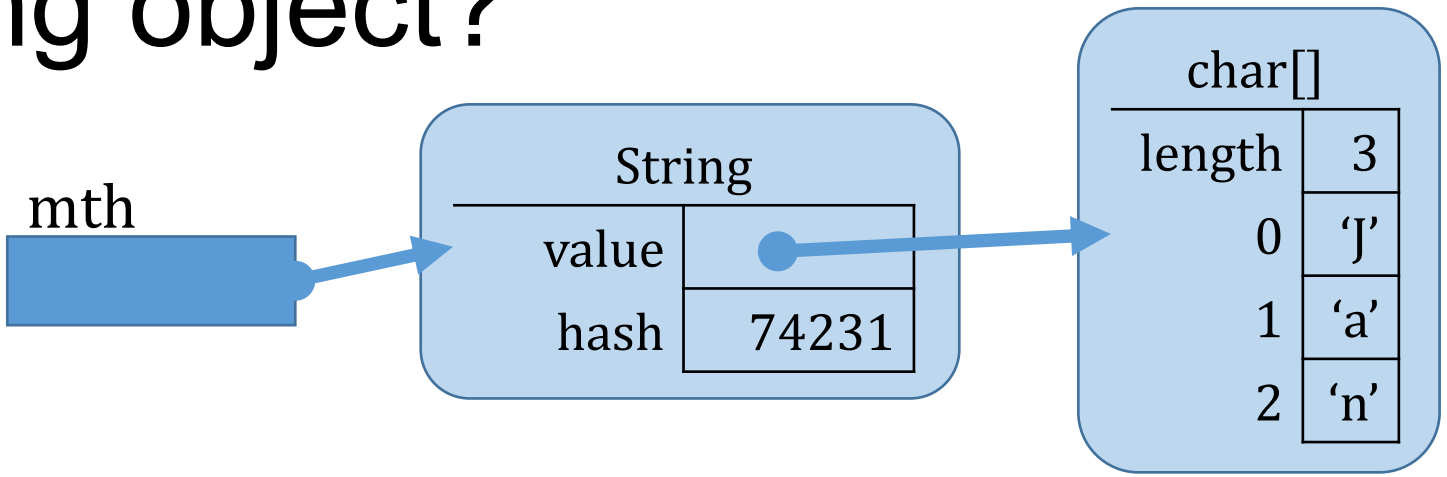
- Make "Point" an immutable class
- Modify Rectangle and Circle to reference new modified Point in "move"
- May be overkill… duplicates objects

Schematic Shared Reference



What's in a String object?

String mth = "Jan";



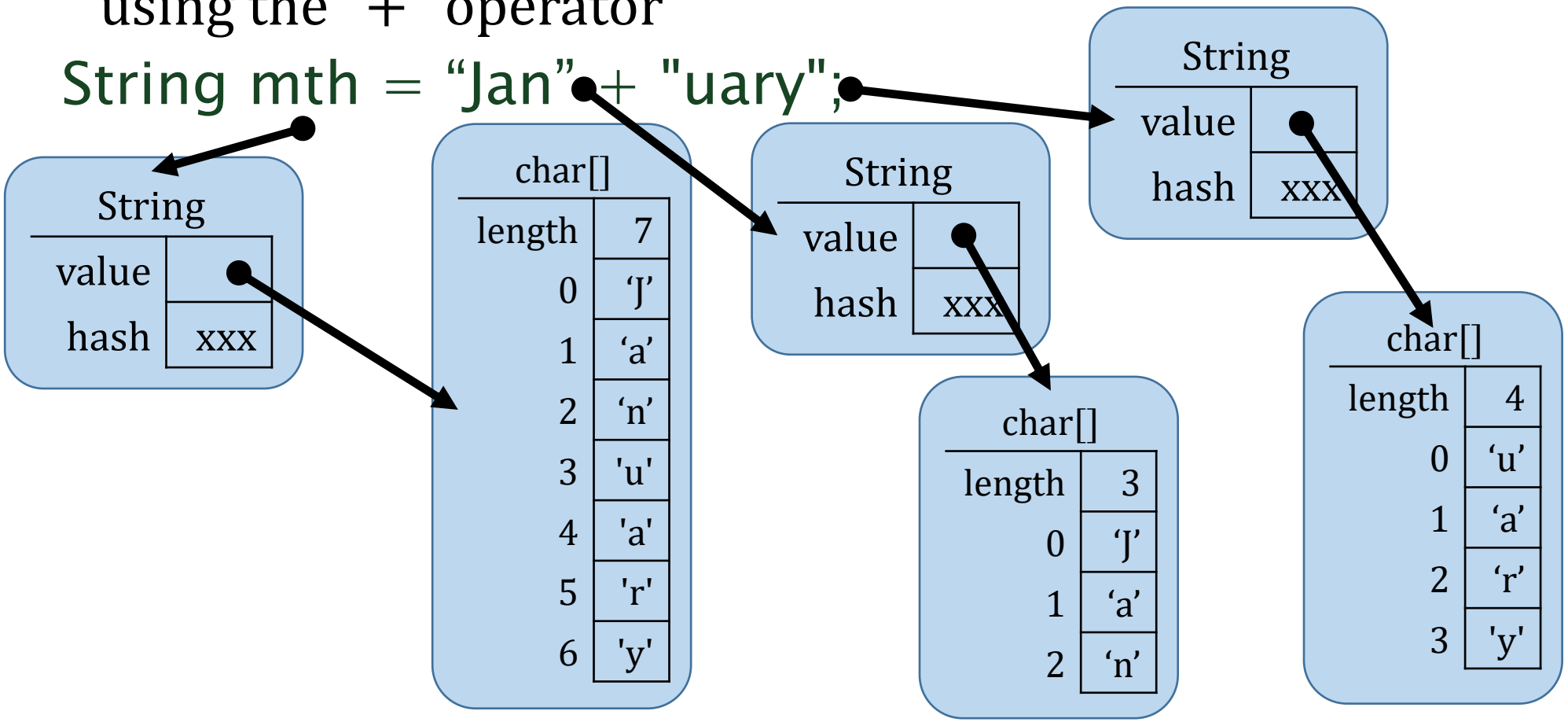
- String is an class with two fields…
  - value : Reference to a character array
  - hash : A numeric value that encodes information about the string into a long integer.

    If two strings have different hash values, the strings are different.

String Concatenation

- Java allows two strings to be concatenated to return a third string using the "+" operator

String mth = “Jan” + "uary";



Concatenation Overhead

- "(" + x + "," + y + ")" Requires objects for
  - "(",
  - "5.5",
  - "(5.5",
  - ",",
  - "(5.5,",
  - "7.0",
  - "(5.5,7.0",
  - ")",
  - "(5.5,7.0)"

That's 9 String objects and 9 character arrays!
8 String objects and 8 character arrays are garbage!

Use "StringBuilder" instead…
a mutable String object with an "append" method