COMP-250 Lecture 8

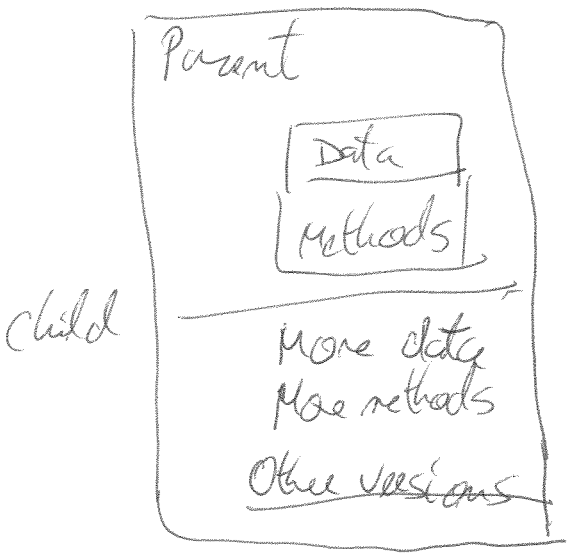
Java - Inheritance ; abstract classes

Parent class

|

Child class extends Parent class

super (something that Parent has)



```
public class PositivePoly extends Poly {
    public PositivePoly() {
        super(); // call to parent constructor
    }
    public PositivePoly(double[] a) throws Exception {
        super(a); // Poly(a) → allocates coefficients
                  //        → copy elements from a
        for (int i=0; i < coefficients.length; i++)
            if (coefficients[i] < 0)
                throw new Exception("No neg coef");
    }
    // similarly
    public PositivePoly(Poly p) throws Exception {
        super(p); // Poly(p)
        // check as above
    }
}
```

Poly → p.add(q)

PositivePoly → p.add(q)

// In PositivePoly

public PositivePoly add(PositivePoly q) {

    Poly r = super.add(q);

look at parent methods; call add

// Children can be used as variables of their parent type

// NOT the other way around!

    return new PositivePoly(r);

result of parent add was a Poly need to "convert" it to PositivePoly

}

// alternatively

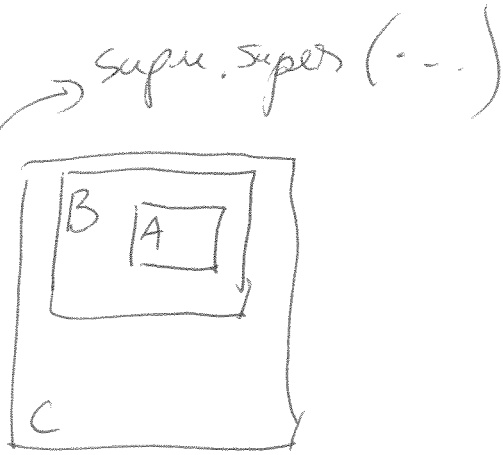
return new PositivePoly(super.add(q));

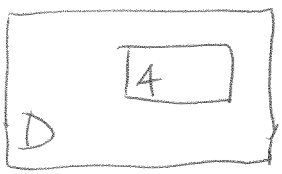
```
public class TestPoly {
    ... main ... {
        PositivePoly p = new PositivePoly({1,2,3});
        PositivePoly q = new PositivePoly({4,5});
        PositivePoly r = p.add(q);
        r.displayPoly(); // same effect as for any Poly
                         // print coef. array
    ...
}
```

What extensions are allowed in Java

```
public class A {...}
public class B extends A {...}
public class C extends B {...}
public class D extends A {...}
public class E extends B, D {...}
```

In Java this is not allowed!!

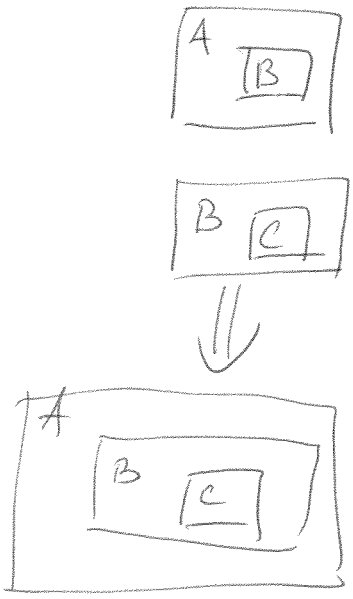




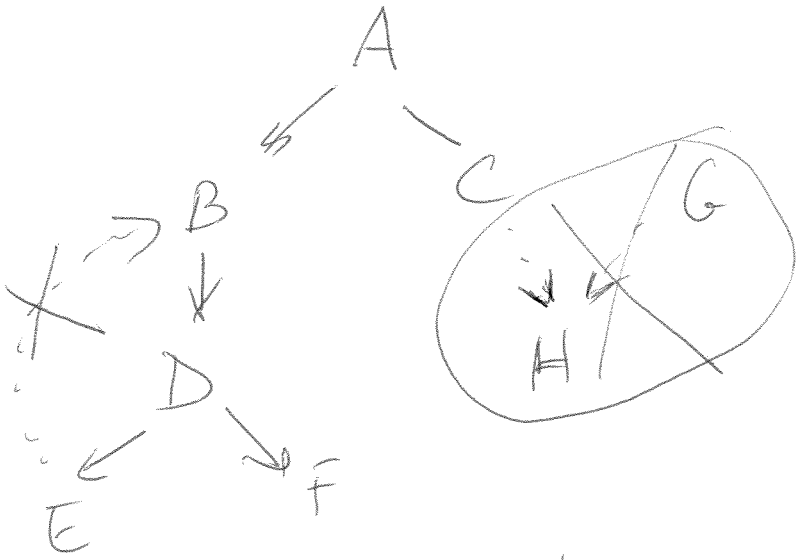
May be naming conflicts

```
public class A extends B {...}
public class B extends C {...}
public class C extends A {...}
```

↳ compiler error!



circular (class would end up depending on itself)



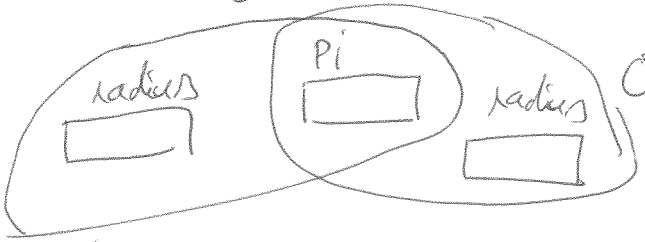
Abstraction → Circular objects

Circular class

- some things are known: all such objects will need a radius, Pi
- some things are not known: exact formula to compute area.

```
public abstract class Circular {
not everything will be specified now
    double radius;
    // Pi is a constant; every body shares it
    // no one can re-declare it
    static final double Pi = 3.14...;
```

static ↑ shared among; final ↑ all objects of this class; no children can re-declare/modify



o_1 o_2

```
    // constructor
    // accessor / mutator for radius
    public abstract double getArea();
```

↑ this method will be written by children

```
public class Circle extends Circular {
    // radius
    // constructor

    public double getArea () {
        return PI * radius * radius;
    }
    // MUST provide an implementation for all
      abstract methods
    // same signature for methods
    // else this class would be abstract too
}
public class Sphere extends Circular {

    public double getArea () {

    }
```