Class relations: has-a

Victor Eijkhout, Harika Gurram,
Je'aime Powell, Charley Dey

Fall 2018

Has-a relationship

A class usually contains data members. These can be simple types or other classes. This allows you to make structured code.

```
class Course {
private:
  Person the_instructor;
  int year;
}
class Person {
  string name;
  ....
}
```

This is called the *has-a relation*.

Literal and figurative has-a

A line segment has a starting point and an end point.

A `Segment` class can store those points:

```
class Segment {
private:
  Point starting_point,ending_point;
public:
  Point get_the_end_point() {
    return ending_point; };
}
  ...
  Segment somesegment;
  Point somepoint =
    somesegment.get_the_end_point();
```

or store one and derive the other:

```
class Segment {
private:
  Point starting_point;
  float length,angle;
public:
  Point get_the_end_point() {
    /* some computation from the
       starting point */ };
}
```

Implementation vs API: implementation can be very different from user interface.

Exercise 1

First implementation:

```
class Rectangle {
private:
  Point bottom_left = Point(0.,0.), top_right = Point(0.,0.);
  bool defined{false};
public:
  Rectangle(Point bl,Point tr) {
    bottom_left = bl; top_right = tr;
    defined = true;
  };
  Rectangle(Point bl,float w,float h) {
    bottom_left = bl; top_right = Point( bl.get_x()+w, bl.get_y()+h );
    defined = true;
  };
  float area() {
    float xsize = top_right.get_x()-bottom_left.get_x();
    float ysize = top_right.get_y()-bottom_left.get_y();
    return xsize*ysize;
  };
};
```

Second implementation:

```
class Rectangle {
private:
  Point bottom_left = Point(0.,0.);
  float width{-1},height{-1};
  bool defined{false};
p[illegible]
  [illegible]angle(Point bl,Point tr) {
    bottom_left = bl;
```

Polymorphism in constructors

You have to decide what to store and what to derive, but you can construct two ways:

```
class Segment {
private:
 // up to you how to implement!
public:
  Segment( Point start,float length,float angle )
    { .... }
  Segment( Point start,Point end ) { ... }
```

Advantage: with a good API you can change your mind about the implementation without bothering the user.