Problem Statement: Draw a house
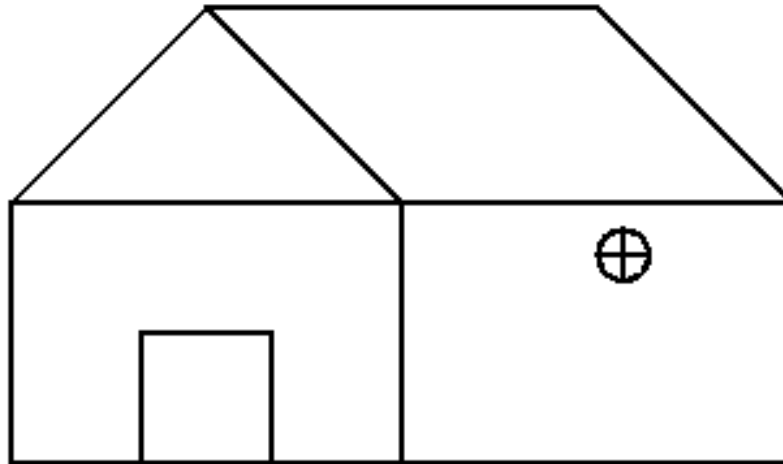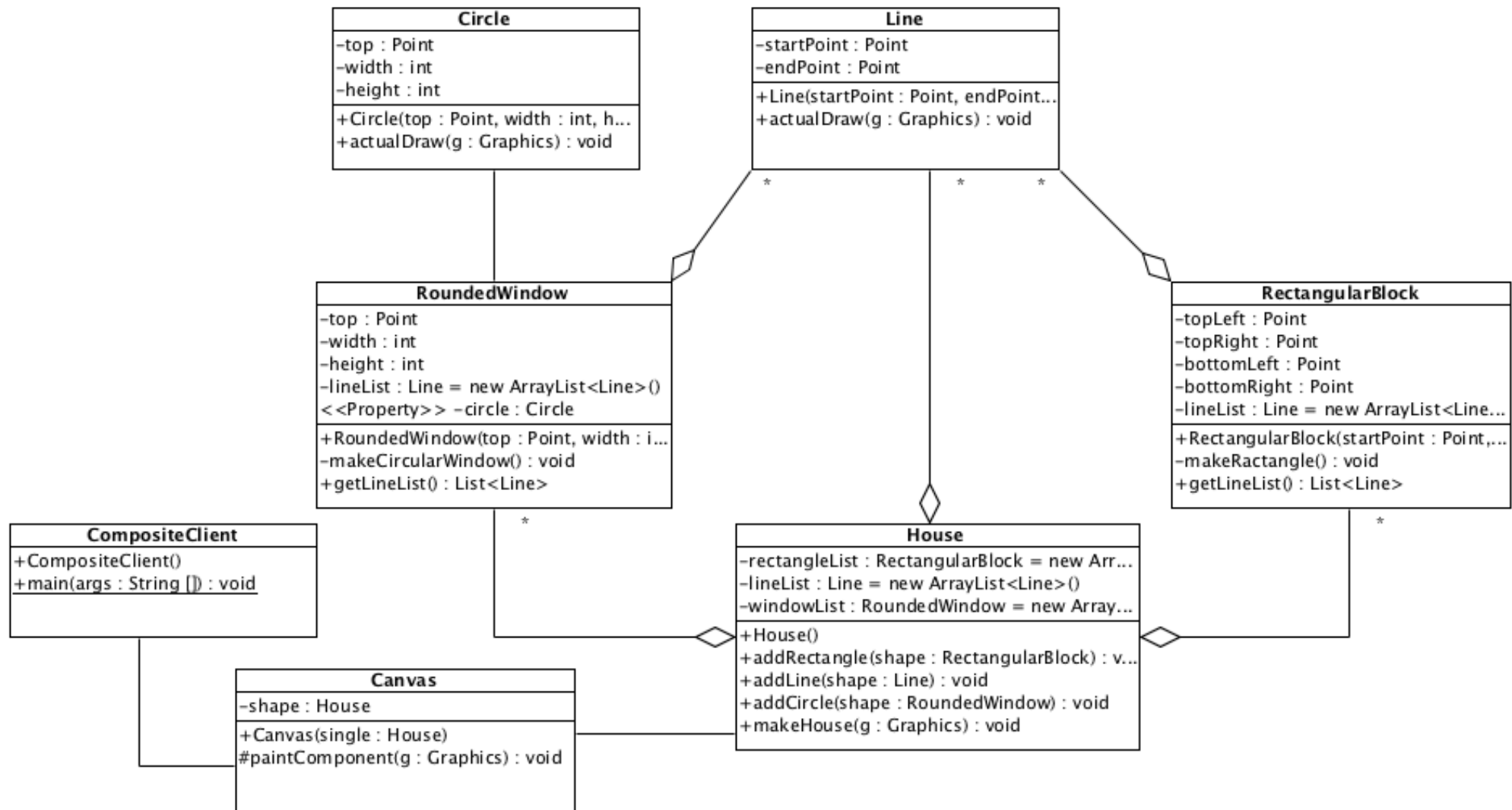
- Draw a house consisting of lines,
  round windows and rectangular blocks

Old (bad) solution in the legacy system

Circle
-top : Point
-width : int
-height : int

+Circle(top : Point, width : int, h...
+actualDraw(g : Graphics) : void

Line
-startPoint : Point
-endPoint : Point

+Line(startPoint : Point, endPoint...
+actualDraw(g : Graphics) : void

RoundedWindow
-top : Point
-width : int
-height : int
-lineList : Line = new ArrayList<Line>()
<<Property>> -circle : Circle

+RoundedWindow(top : Point, width : i...
-makeCircularWindow() : void
+getLineList() : List<Line>

RectangularBlock
-topLeft : Point
-topRight : Point
-bottomLeft : Point
-bottomRight : Point
-lineList : Line = new ArrayList<Line...

+RectangularBlock(startPoint : Point,...
-makeRactangle() : void
+getLineList() : List<Line>

CompositeClient
+CompositeClient()
+main(args : String []) : void

House
-rectangleList : RectangularBlock = new Arr...
-lineList : Line = new ArrayList<Line>()
-windowList : RoundedWindow = new Array...

+House()
+addRectangle(shape : RectangularBlock) : v...
+addLine(shape : Line) : void
+addCircle(shape : RoundedWindow) : void
+makeHouse(g : Graphics) : void

Canvas
-shape : House

+Canvas(single : House)
#paintComponent(g : Graphics) : void

Task #1: Use the Composite pattern (15 min)

- Problem in the code of the legacy system:
  - RoundedWindow and the RectangularBlock don't have a draw() method

  - The House class knows how to draw the house (non-flexible implementation)

  - **Your Task :** Refactor the legacy code by using the composite pattern.

```java
public void makeHouse(Graphics g) {
    for (RoundedWindow item : windowList) {
        item.getCircle().actualDraw(g);
        for (Line lineItem : item.getLineList()) {
            lineItem.actualDraw(g);
        }
    }
    for (RectangularBlock item : rectangleList) {
        for (Line lineItem : item.getLineList())
            lineItem.actualDraw(g);
    }
    for (Line lineItem : lineList) {
        lineItem.actualDraw(g);
    }
}
```