

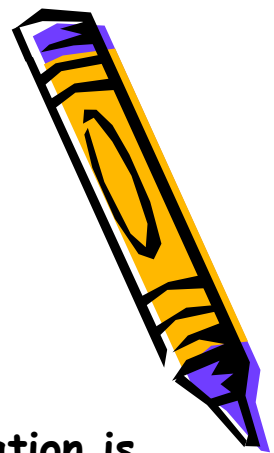


Transformations

java Graphics2D, translate(),
rotate() & scale()
#2



translate(), rotate() & scale()

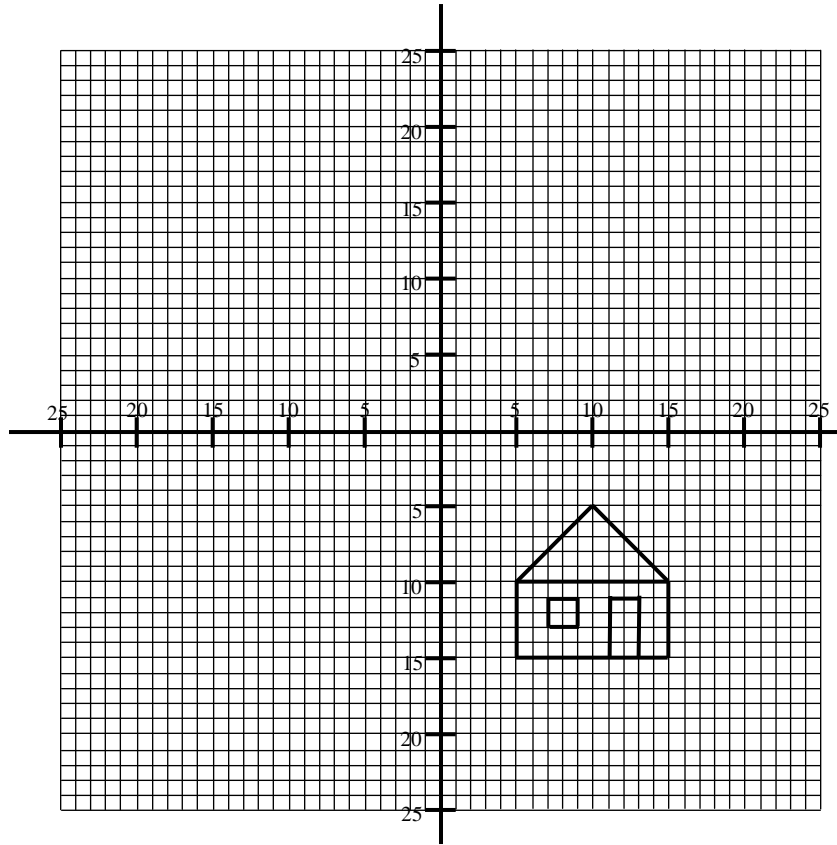
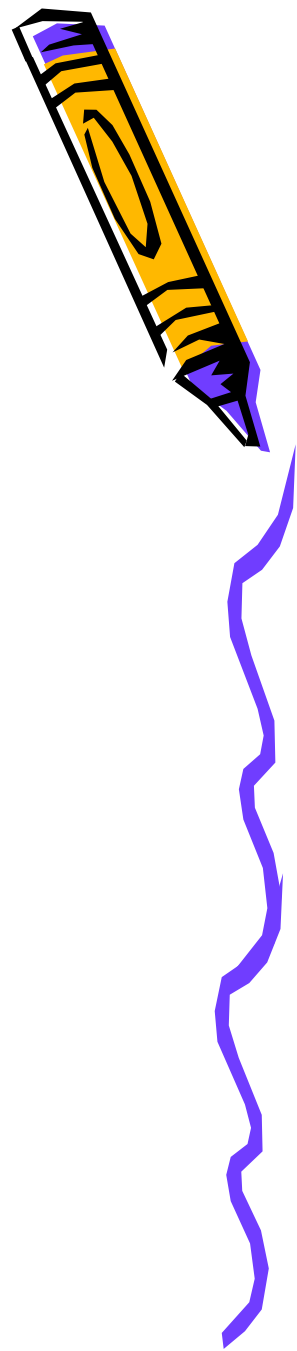


Unlike previous examples the best way to think about a transformation is to thinking of the **effect on the shape** NOT the **point of origin**. This requires you to move any point of the shape to the origin do any transformations and then move to final point. In doing this write the first thing you do as the last instruction, the second thing you do as the second last instruction and so on. In other words you need to write the transformation steps in **reverse order**, e.g. these are the final instructions created by the backward construction technique.

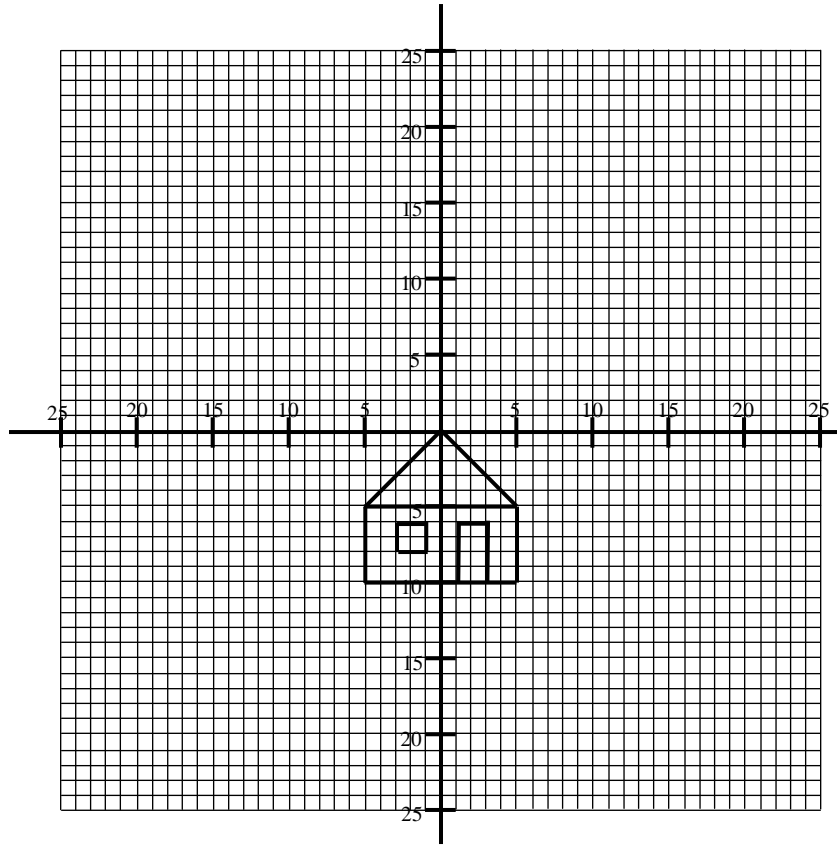
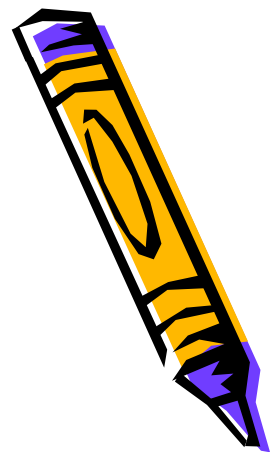
```
g2.translate(5, 10);  
g2.rotate(Math.toRadians(-135));  
g2.scale(0.5, 0.5);  
g2.scale(-1, 1);  
g2.translate(-10, -5);  
g2.draw(s);
```



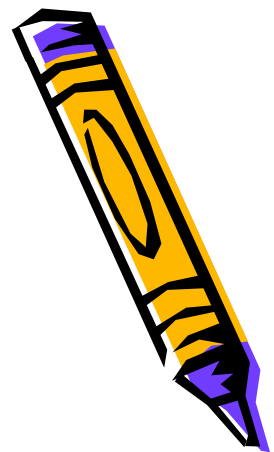
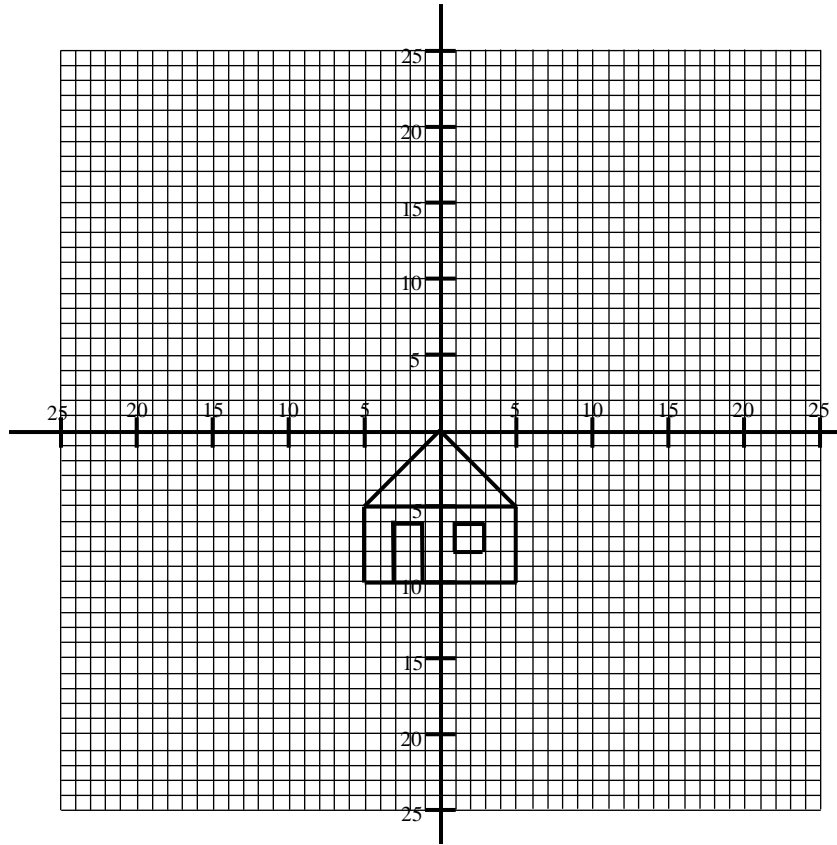
Starting Image



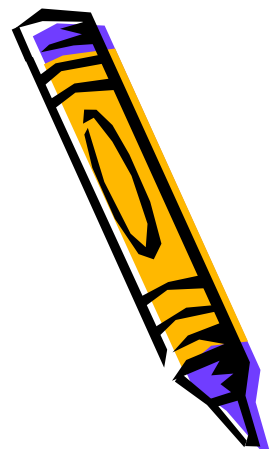
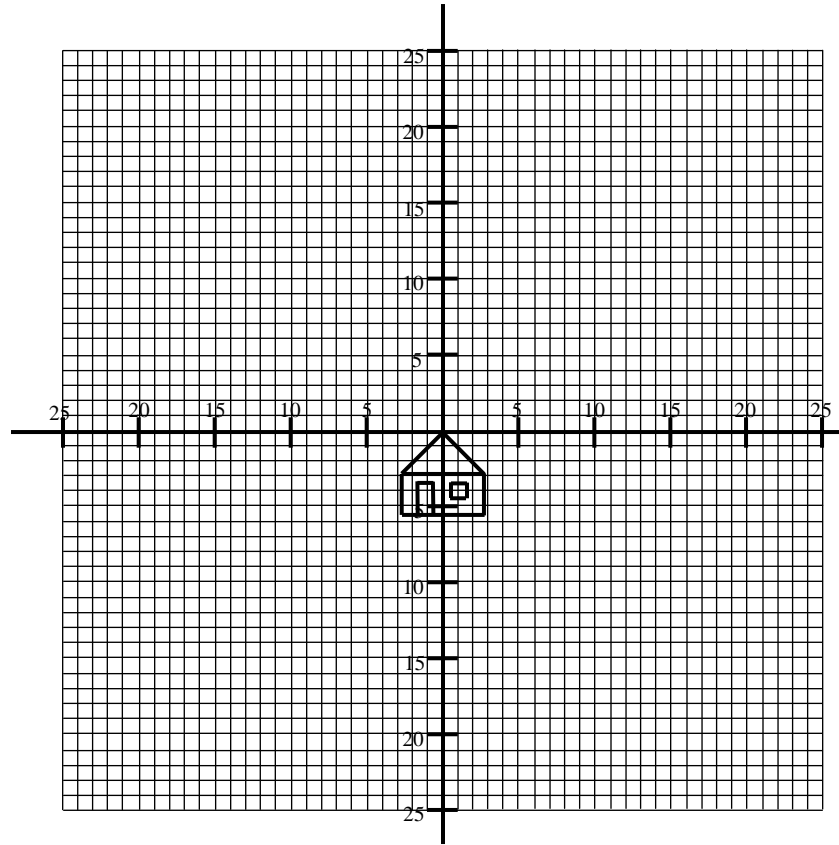
`g2.translate(-10, -5);`



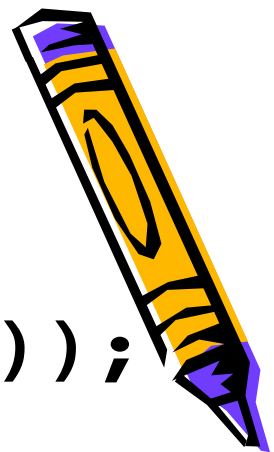
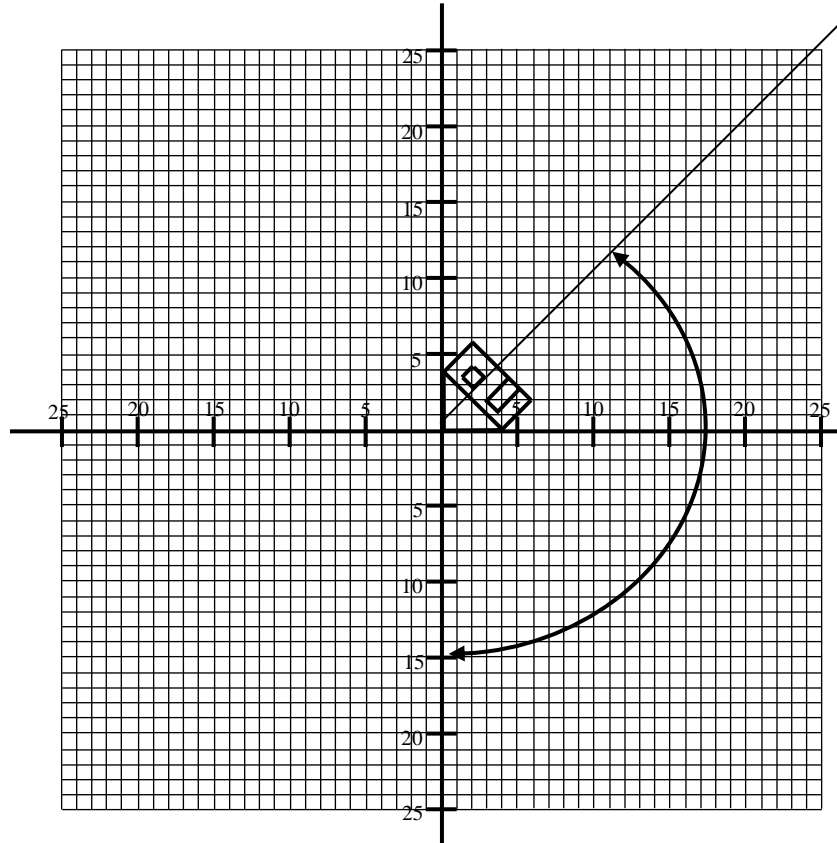
```
g2.scale(-1, 1);
```



```
g2.scale(0.5, 0.5);
```



```
g2.rotate(Math.toRadians(-135));
```



```
g2.translate(10, 5);
```

