

# Computer Graphic Assignment 2

Xiang Zhang    CS560

The assignment is designed for polygon clipper, filling and transformation. The algorithm I implemented to clip the polygon is Sutherland-Hodgeman algorithm.

To run the code: `g++ assign2.c -o assign2 -framework OpenGL -framework GLUT`

1. All the codes are in `assign2.c`, in which I implement the Sutherland-Hodgeman algorithm to clip a polygon. The method is :

```
void suthHodgClipping(Point points[], Point  
clipPoints[], int *number) ;
```

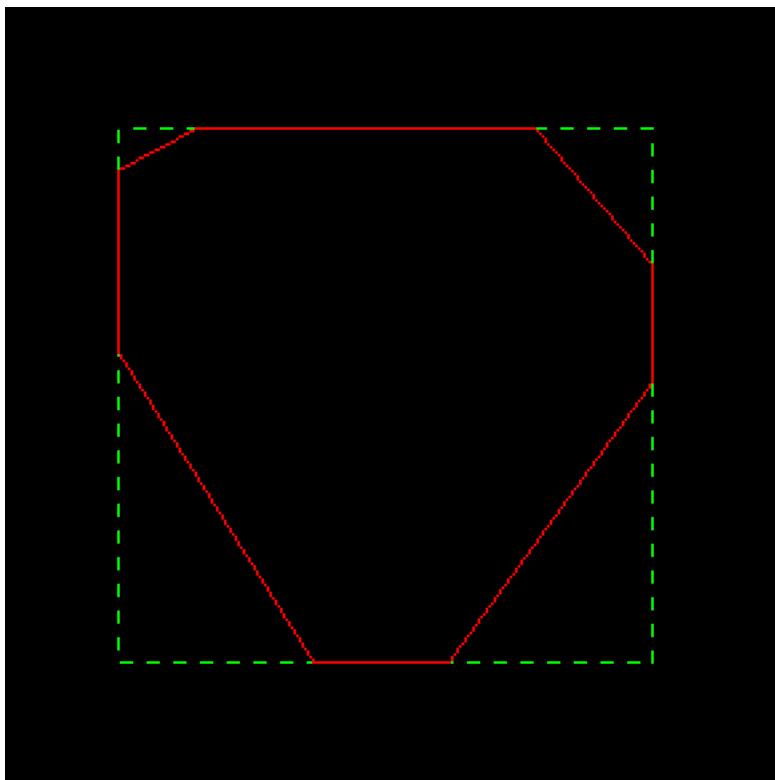
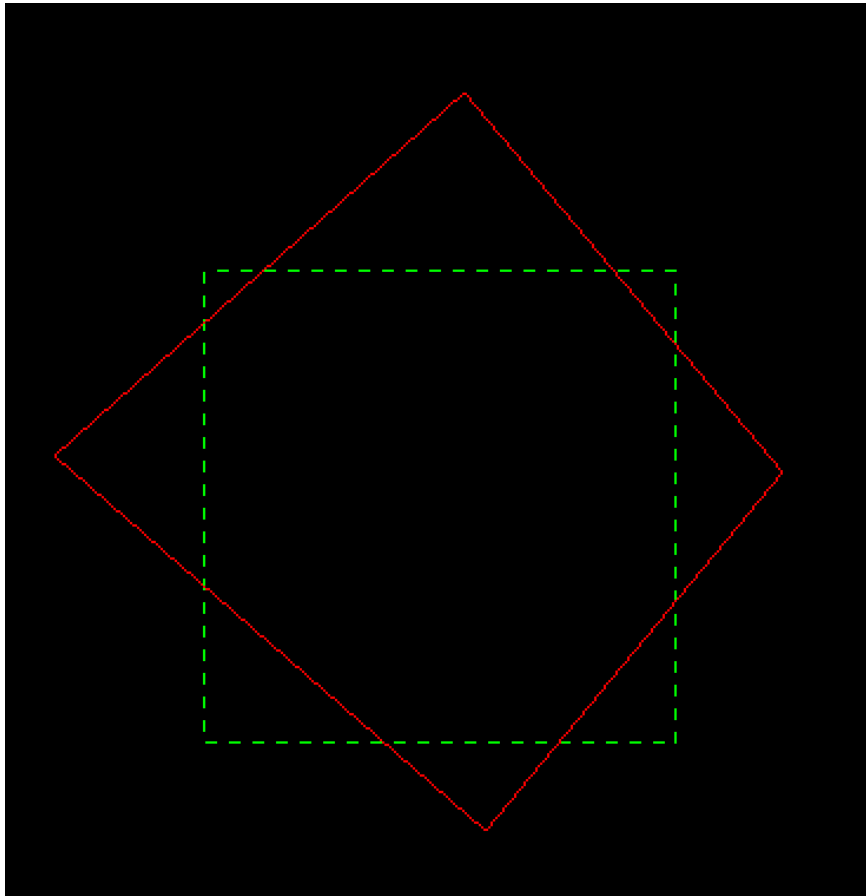
which contains the clipping method for left, right, up and bottom boundary.

```
void clipping(Point points[], Point  
clipPoints[], LinePosition linePosition, int  
*number)
```

Before that, I used the global variable `_polygon` of Polygon structure to store the points of polygon, which will be draw through line by OpenGL method. By clipping method, the polygon's points will be updated.

Left mouse clicking to determine the vertex and right mouse clicking finish drawing it. Then show the menu by right click. Choose polygon clicking

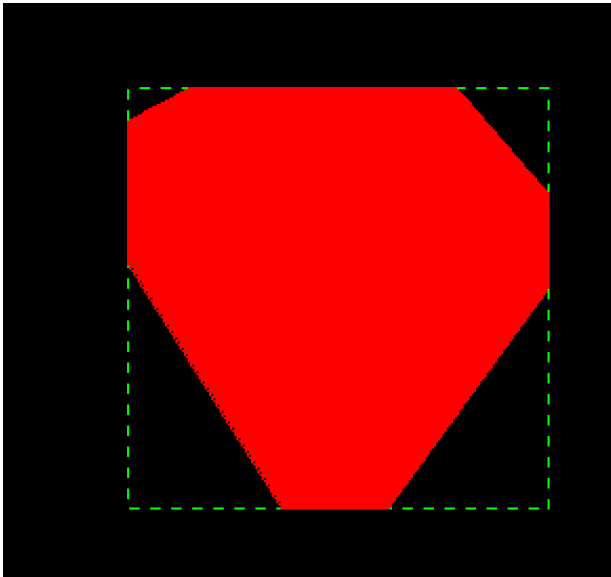
to clip the polygon with Sutherland-Hodgeman algorithm.



2. `void scanFilling()` is the filling method to fill color of the polygon. I try to used flood and scan-line algorithms to implement this, but there's a problem of `glReadPixels()` method. So I using a method to scan all the pixels within the polygon and windows to set color of them.

The relate method is: `bool insidePolygon(Point point)`

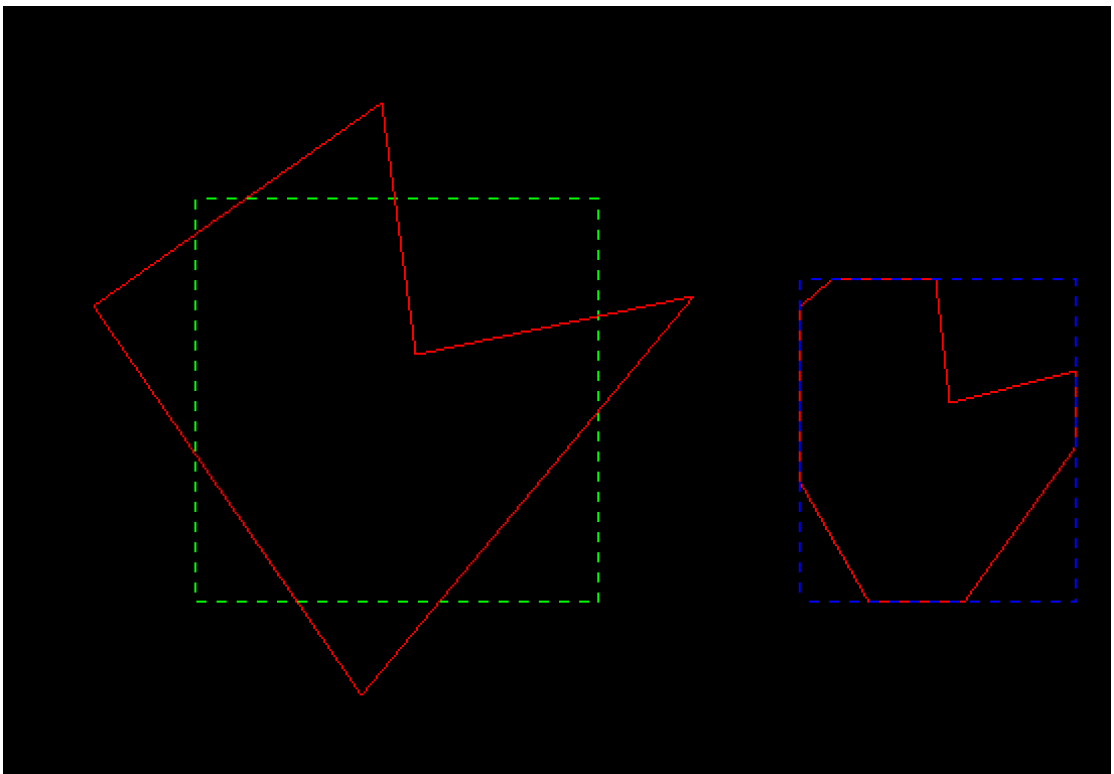
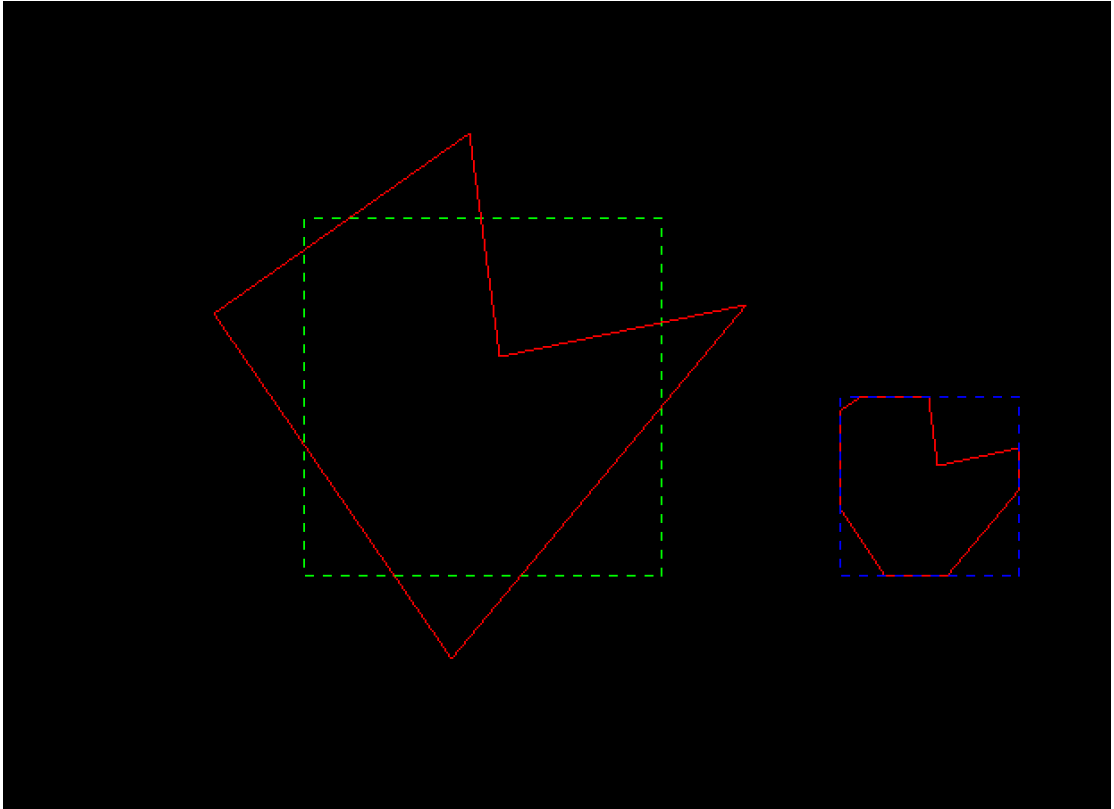
Filling method is in the right click menu :



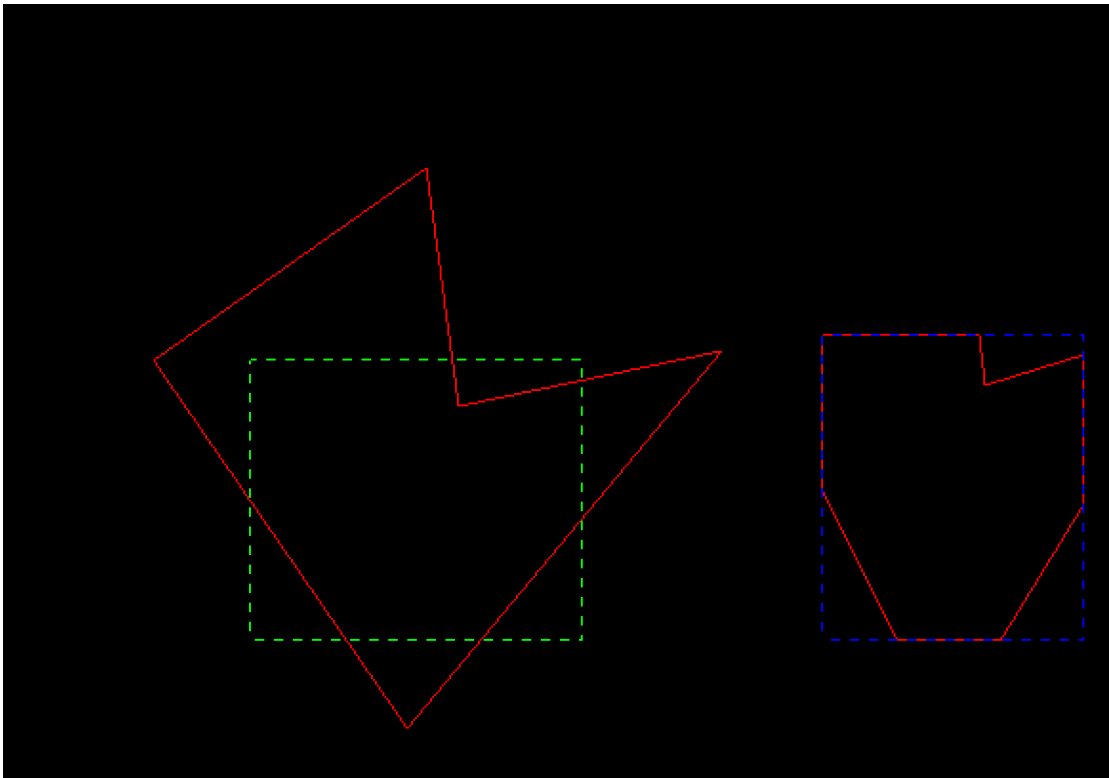
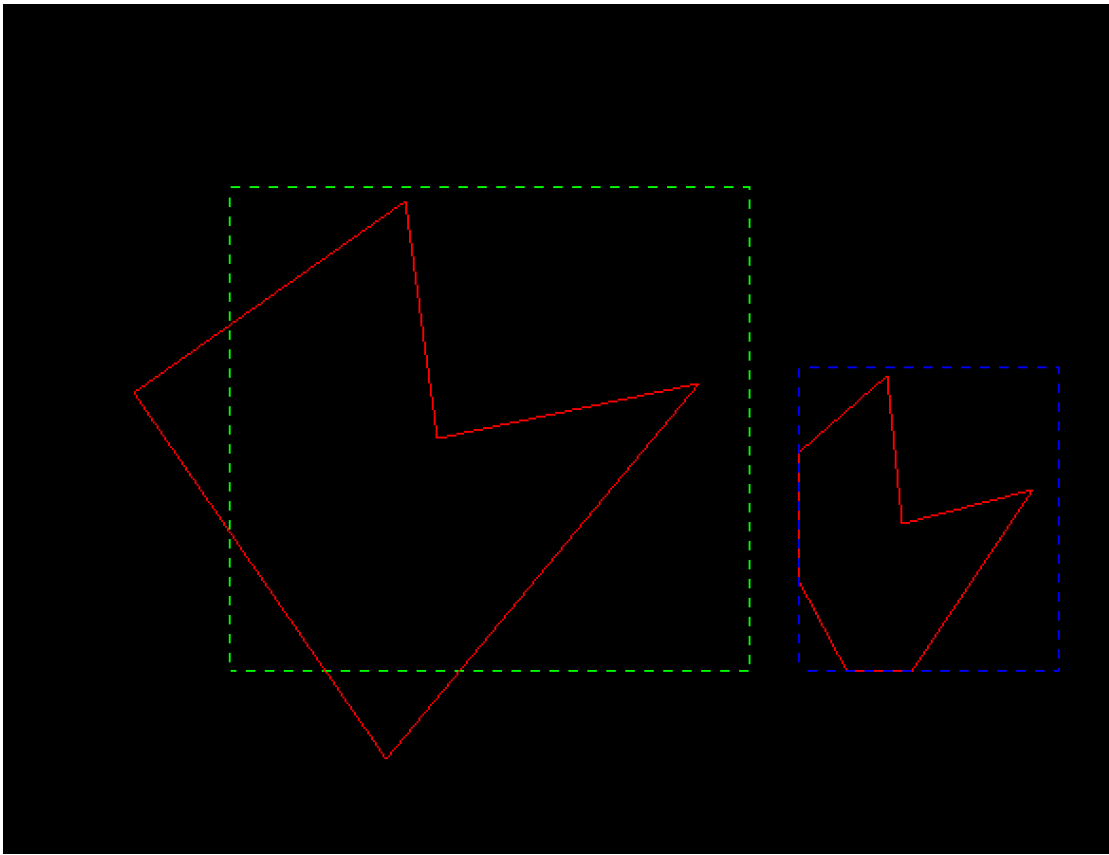
3. To translate the view to viewport, we need to make the matrix first.

`void makeMatrix()`

is the method of making matrix. Dragging the right up corner will effect the matrix. The four vertex points of view port are stored in `_viewPoints` variables.



Meanwhile, the matrix will also be changed by dragging the right up corner, which implement zoom in, zoom out effect.



Last, dragging the window with the mouse within the window could move the window freely. That's also change the variables within `makematrix()`.

