Jason Tieu

304047667

CS174A

Dis 1A

***Assignment 1 Part 1***

1.

A:

B:

C:

2.

3.

modelMatrix.setAsIdentity();

modelMatrix \*= Translate(1,1,1);

modelMatrix \*= Scale(1,1,2);

4.

5.

A: B: C: D:

6.

***For arbitrary tilted line y = mx + b***

M =

M = Translate(0,b,0) \* RotateX() \* Scale(1,-1,1) \* RotateX(-) \* Translate(0,-b,0)

***For arbitrary tilted line y = x - 1 and = 45***

In OpenGL Shader code:

modelMatrix.setAsIdentity();

modelMatrix \*= Translate(0,b,0);

modelMatrix \*= RotateX();

modelMatrix \*= Scale(1,-1,1);

modelMatrix \*= RotateX(-);

modelMatrix \*= Translate(0,-b,0);

***For arbitrary tilted line y = x - 1 and = 45***

M =

M =

***For arbitrary tilted line y = x - 1***

In OpenGL Shader code:

modelMatrix.setAsIdentity();

modelMatrix \*= Translate(0,-1,0);

modelMatrix \*= RotateX();

modelMatrix \*= Scale(1,-1,1);

modelMatrix \*= RotateX();

modelMatrix \*= Translate(0,1,0);

7.

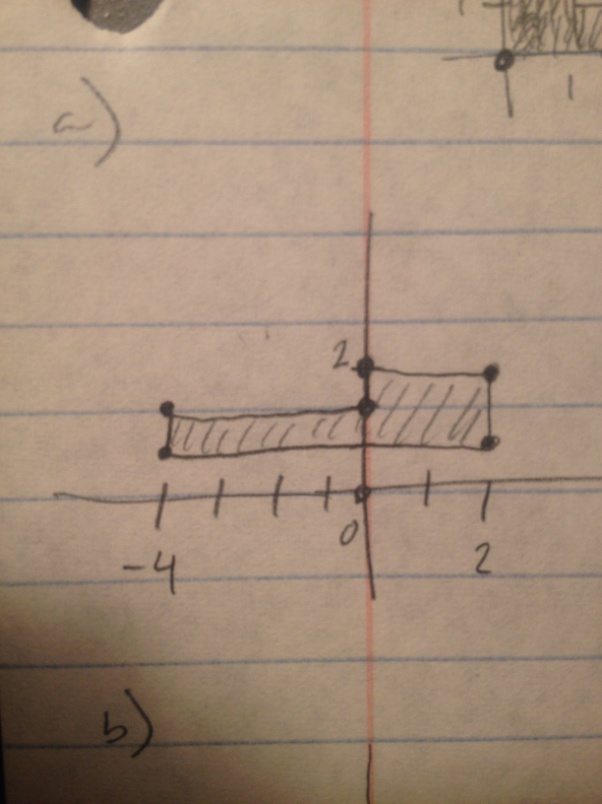
a)L' = ABCL

modelMatrix \*= Scale(2,1,1);

modelMatrix \*= Translate(1,1,0);

modelMatrix \*= RotateZ(90);

drawL();



b) L' = CADL

modelMatrix = matrixStack.pop();

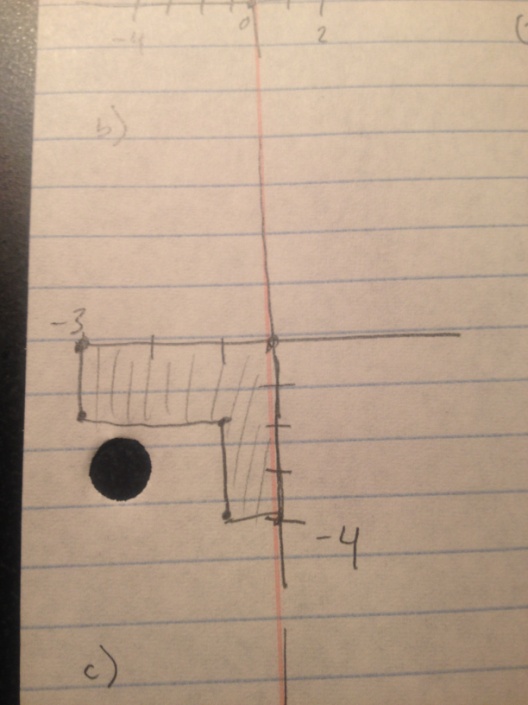
matrixStack.push(modelMatrix); //saves for part c

modelMatrix \*= RotateZ(90);

modelMatrix \*= Scale(2,1,1);

modelMatrix \*= Scale(-1,1,1);

drawL();



c) L' = CBDL

modelMatrix = matrixStack.pop();

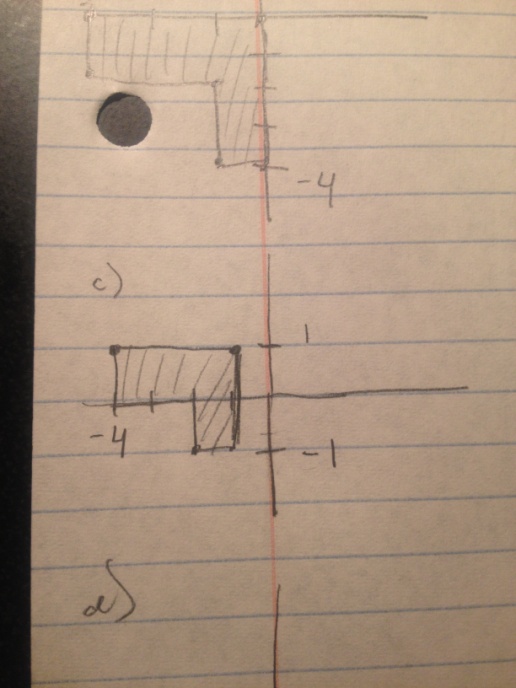
matrixStack.push(modelMatrix); //saves for part d

modelMatrix \*= RotateZ(90);

modelMatrix \*= Translate(1,1,0);

modelMatrix \*= Scale(-1,1,1);

drawL();



d) L' = DCCADL

modelMatrix = matrixStack.pop();

modelMatrix \*= Scale(-1,1,1);

modelMatrix \*= RotateZ(90);

modelMatrix \*= RotateZ(90);

modelMatrix \*= Scale(2,1,1);

modelMatrix \*= Scale(-1,1,1);

drawL();

