Write a program to implement the slope independent line drawing algorithm.

Steps:

1. Generate coordinates for a line AB. A(x0,y0) and B(x1,y1)
2. Determine the zone of the line [0 … 7]
   1. If(|dx|>|dy|) [zones 0,7,3,4]
      1. If(dx>0 && dy>0)
         1. Return Zone=0
      2. Else if (?)
         1. Return zone=7
      3. Else if (?)
         1. Return zone=3
      4. Else
         1. Return zone=4
   2. Else [zones 1,2,5,6
      1. If(?)
         1. Return Zone=1
      2. Else if (?)
         1. Return zone=2
      3. Else if (?)
         1. Return zone=5
      4. Else
         1. Return zone=6
3. Calculate next pixel for zone 0
   1. If (zone 1|| 2||5||6)
      1. Swap (x0,y0)
      2. Swap (x1,y1)
   2. Dinit=2dy-dx
   3. dNE=?
   4. dE=?
   5. Loop until (x0<x1)
      1. If(dinit<0)
         1. Dinit+=dE
      2. Else
         1. Dinit+=dNE
         2. Y++
      3. X++
      4. Convert back to correct zones coordinates **ConvBack(x,y)**
4. **ConvBack(x,y){ //each zone should have a different color**
   1. If(zone==0)

gl.glColor3f(1f,0f,0f);

* + 1. drawPixel(x,y)
  1. else if (zone==7)
     1. drawPixel(x,-y)
  2. else if(zone==3)
     1. drawPixel(?,?)
  3. else if(zone==4)
     1. drawPixel(?,?))
  4. else if(zone==1)
     1. drawPixel(?,?)
  5. else if(zone==2)
     1. drawPixel(?,?)
  6. else if(zone==3)
     1. drawPixel(?,?)
  7. else if(zone==4)
     1. drawPixel(?,?)

**}**

gl.glColor3f(1f,0f,0f); //gives us red

gl.glColor3f(0f,1f,0f); //gives us green

gl.glColor3f(0f,0f,1f); //gives us blue