```
#include<stdio.h>
#include <GL/gl.h>
#include <GL/glut.h>
float x1,y1,x2,y2,m,i,j;
float dx,dy;
void display(void)
{

   glClear (GL_COLOR_BUFFER_BIT);

   glEnd();

   glColor3f (0.0, 0.0, 1.0);
   glBegin(GL_POINTS);



   if(m > 0 && m <= 1)
   {
      while(x1 <= x2 && y1 <= y2)
      {
         x1 = x1 + 1;
         y1 = y1 + m;
         glVertex3f(x1,y1,0.0);
         printf("%f %f",x1,y1);

      }
   }
```

```
else if(m > 1)
{
   while(x1 <= x2 && y1 <= y2)
   {
      x1 = x1 + (1/m);
      y1 = y1 + 1;
      glVertex3f(x1,y1,0.0);
      printf("%f %f",x1,y1);
   }
}

else if(m > -1 && m <= 0)
{
   while(x1 >= x2 && y1 >= y2)
   {
      x1 = x1 - 1;
      y1 = y1 - m;
      glVertex3f(x1,y1,0.0);
      printf("%f %f",x1,y1);
   }
}
else if(m < -1)

{

   while(x1 >= x2 && y1 >= y2)
   {
```

```
        x1 = x1-(1/m);
        y1 = y1-1;
        glVertex3f(x1,y1,0.0);
        printf("%f %f",x1,y1);
    }
}

glEnd();

// Triangle
glBegin(GL_TRIANGLES);
glColor3f (0.0, 0.0, 1.0);

glVertex3f(15.0f, 5.0f, 0.0f);
glVertex3f(25.0f, 5.0f, 0.0f);
glVertex3f(20.0f, 12.0f, 0.0f);
glEnd();

//Trapezoid

glBegin(GL_QUADS);
glColor3f (1.0, 0.0, 0.0);

glVertex3f(5.0f, 18.0f, 0.0f);
glVertex3f(15.0f, 18.0f, 0.0f);
glVertex3f(15.0f, 27.0f, 0.0f);
glVertex3f(5.0f, 27.0f, 0.0f);
```

```
    glEnd();



    glFlush ();


}
void init (void)
{
    glClearColor (1.0, 1.0, 1.0, 0.0);
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    glOrtho(0.0, 30.0, 0.0, 30.0, -1.0, 1.0);
}

int main(int argc, char** argv)
{
    x1 = 0, y1 = 0, x2 = 30, y2 = 30;
    dx = x2-x1;
    dy = y2-y1;
    m = dy/dx;



    glutInit(&argc, argv);
    glutInitDisplayMode (GLUT_SINGLE | GLUT_RGB);
```

```
    glutInitWindowSize (500, 500);
    glutInitWindowPosition (100, 100);
    glutCreateWindow ("Lab Performance II");
    init ();
    glutDisplayFunc(display);
    glutMainLoop();
    return 0;
}
```