**Assignment number: 8**

**Subject: COMPUTER GRAPHICS LAB**

Name: ***RIA MITTAL***

Class: ***SECOND YEAR ENGINEERING***

Division: ***B***

Roll no: ***222008***

Batch: ***B1***

**PROBLEM STATEMENT:**

Write a Java program to fill polygon using scan line algorithm.

**Code:**

package scan;

import java.awt.Color;

import java.awt.Graphics;

import static java.awt.Color.RED;

import javax.swing.JFrame;

public class Scan extends JFrame {

public static void main(String[] args){

Scan m= new Scan();

m.setTitle("Draw");

m.setSize(1000,1000);

m.setVisible(true);

m.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

}

double slope(double x2,double y2,double x1,double y1){

return (y2-y1)/(x2-x1);

}

public void paint(Graphics g){

int in[] =new int[10];

int arr [][]=new int [3][2];

///unused coordinates

arr[0][0]=100;

arr[0][1]=320;

arr[1][0]=400;

arr[1][1]=320;

arr[2][0]=250;

arr[2][1]=60;

g.drawLine((int)arr[0][0], (int)arr[0][1],(int) arr[1][0],(int) arr[1][1]);

g.drawLine((int)arr[1][0],(int)arr[1][1],(int)arr[2][0],(int) arr[2][1]);

g.drawLine((int)arr[0][0],(int)arr[0][1],(int)arr[2][0],(int) arr[2][1]);

double m1,m2,m3;

m1=slope(arr[1][0],arr[1][1],arr[0][0],arr[0][1]);

m2=slope(arr[2][0],arr[2][1],arr[1][0],arr[1][1]);

m3=slope(arr[2][0],arr[2][1],arr[0][0],arr[0][1]);

int ymin=arr[0][1];

int ymax=arr[0][1];

int imax,imin,mid;

imax=0;

imin=0;

mid=0;

double xl,xr;

for(int i=0;i<3;i++){

if(ymin>arr[i][1]){

ymin=arr[i][1];

imin=i;

}

if(ymax<arr[i][1]){

ymax=arr[i][1];

imax=i;

}

}

for(int i=0;i<3;i++){

if(i!=imax&&i!=imin)

mid=i;

}

double mtarget=slope(arr[imax][0],arr[imax][1],arr[imin][0],arr[imin][1]);

double ml=slope(arr[imin][0],arr[imin][1],arr[mid][0],arr[mid][1]);

double mh=slope(arr[mid][0],arr[mid][1],arr[imax][0],arr[imax][1]);

System.out.println("Mtarget= "+mtarget+" \n ml ="+ml+"\n mh ="+mh);

int cnt=0;

for(int i=arr[imin][1];i<arr[imax][1];i++){

try {

Thread.sleep(20);

} catch (InterruptedException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

g.setColor(Color.magenta);

if(i<arr[mid][1])

{

xl=arr[imin][0]+((i-arr[imin][1])/mtarget);

xr=arr[imin][0]+((i-arr[imin][1])/ml);

g.drawLine((int)xl,i,(int)xr,i);

}

else{

xl=arr[imin][0]+((i-arr[imin][1])/mtarget);

xr=arr[imax][0]+((i-arr[imax][1])/mh);

g.drawLine((int)xl,i,(int)xr,i);

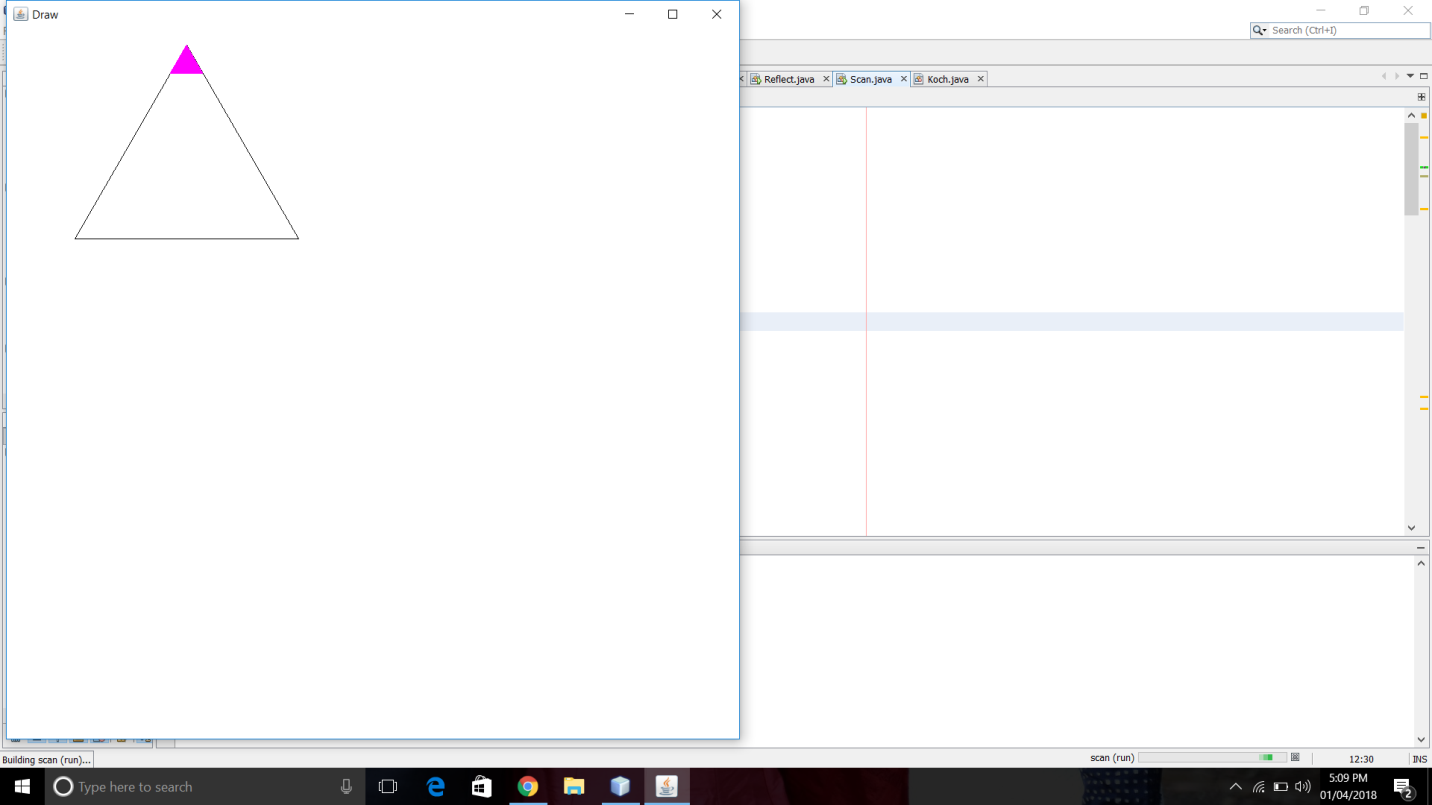
}

}

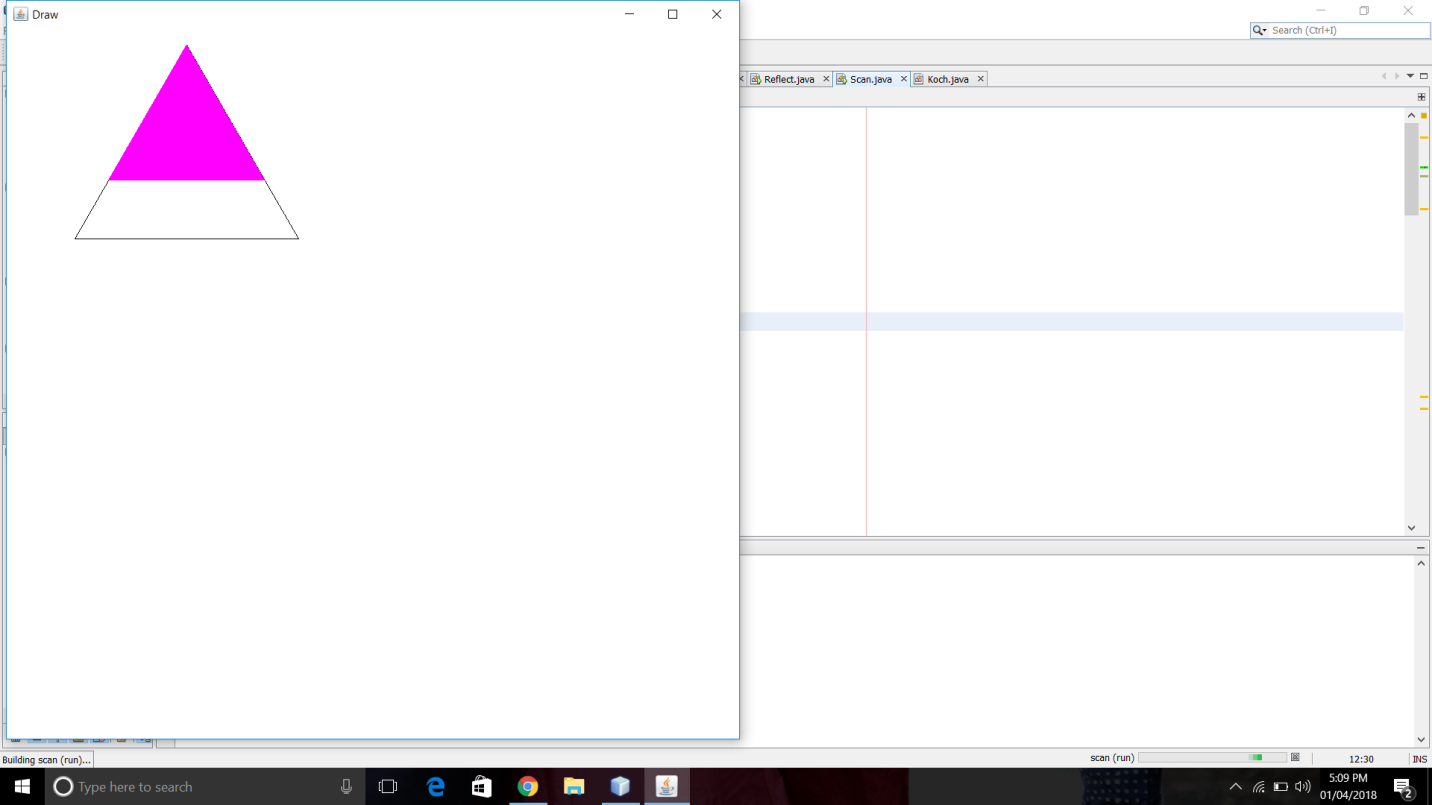
}

}

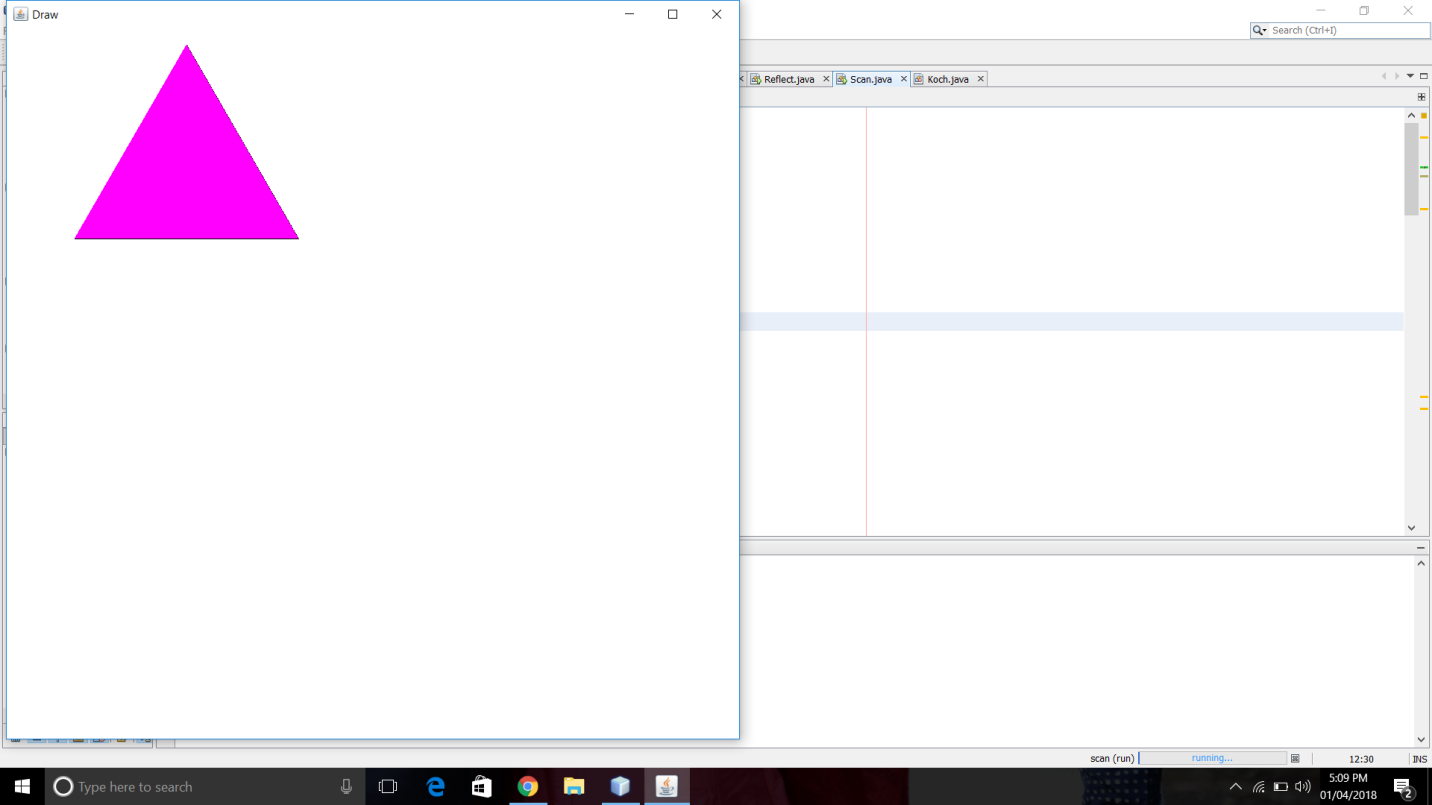
**Starting to fill:**



**Half filled:**

****

**Fully filled:**

****