---------------------------------------------------------------------- Write C++/Java program to implement reflection of2-D object about X axis, Y axis and about X=Y axis. Also rotate object about arbitrary point given by user

----------------------------------------------------------------------#include "refl.h"

#include "ui\_refl.h"

#include<QPainter>

#include "QMouseEvent"

refl::refl(QWidget \*parent) :

QMainWindow(parent),

ui(new Ui::refl)

{

ui->setupUi(this);

}

refl::~refl()

{

delete ui;

}

void refl::paintEvent(QPaintEvent \*)

{

if(flag==0)

{ draw2();

draw();

}

if (flag==1)

{

draw2();

draw();

QScreen \*screen = QGuiApplication::primaryScreen();

QRect screenGeometry = screen->geometry();

height = screenGeometry.height();//1360;//

width = screenGeometry.width();

for(int j=0;j<n;j++)

{

xr[j]=x[j];

yr[j]=height-y[j];

}

draw3();

}

if (flag==2)

{

draw2();

draw();

QScreen \*screen = QGuiApplication::primaryScreen();

QRect screenGeometry = screen->geometry();

height = screenGeometry.height();//1360;//

width = screenGeometry.width();

for(int j=0;j<n;j++)

{

xr[j]=width-x[j];

yr[j]=y[j];

}

draw3();

}

if(flag==3)

{

draw2();

draw();

QScreen \*screen = QGuiApplication::primaryScreen();

QRect screenGeometry = screen->geometry();

height = screenGeometry.height();//1360;//

width = screenGeometry.width();

int d;

int a=(-height/width);

for(int j=0;j<n;j++)

{

d=(x[j]+(y[j] - height)\*a)/(1+a\*a);

xr[j]=2\*d-x[j];

yr[j]=2\*d\*a-y[j]+2\*height;

}

draw3();

}

if(flag==4)

{

draw2();

draw();

QScreen \*screen = QGuiApplication::primaryScreen();

QRect screenGeometry = screen->geometry();

height = screenGeometry.height();//1360;//

width = screenGeometry.width();

for(int j=0;j<n;j++)

{

xr[j]=width-x[j];

yr[j]=height-y[j];

}

draw3();

}

}

void refl:: draw3()

{

QPainter painter(this);

painter.setPen(Qt::red);

i=flag=0;

xr[n]=xr[0];

yr[n]=yr[0];

for(int j=0;j<n;j++)

painter.drawLine(xr[j],yr[j],xr[j+1],yr[j+1]);

}

void refl::mousePressEvent(QMouseEvent \*)

{

if(i<n)

{

x[i]=QCursor::pos().x();

y[i]=QCursor::pos().y();

i++;

}

}

void refl:: draw()

{

QPainter painter(this);

painter.setPen(Qt::blue);

i=flag=0;

x[n]=x[0];

y[n]=y[0];

for(int j=0;j<n;j++)

painter.drawLine(x[j],y[j],x[j+1],y[j+1]);

}

void refl :: draw2()

{

QPainter painter(this);

QScreen \*screen = QGuiApplication::primaryScreen();

QRect screenGeometry = screen->geometry();

height = screenGeometry.

height();//1360;//

width = screenGeometry.width();//768;//

painter.drawLine(0,height/2,width,height/2);

painter.drawLine(width/2,0,width/2,height);

int cnt,x,y;

char buffer[2];

painter.setPen(QPen(Qt::blue,2));

x=40;y=height/2;

for(cnt=-16;cnt<=16;cnt++)

{

painter.setPen(QPen(Qt::blue,2));

sprintf(buffer,"%d",cnt);

painter.drawText(x,y,buffer);

painter.setPen(QPen(Qt::red,3));

painter.drawPoint(x,y);

x=x+40;

}

x=width/2;y=30;

for(cnt=9;cnt>=-9;cnt--)

{

painter.setPen(QPen(Qt::blue,2));

if(cnt!=0)

{

sprintf(buffer,"%d",cnt);

painter.drawText(x,y,buffer);

painter.setPen(QPen(Qt::red,3));

painter.drawPoint(x,y);

}

y=y+40;

}/\*char b[5];

int r,k,s;

s=width/2;

k=0;

for(r=width/2;r>0;r=r-25)

{

sprintf(b,"%d",-k);

//painter.drawText(r-6,337,b);

if(k!=0)

{

sprintf(b,"%d",k);

//painter.drawText(r,337,b);

}

painter.setPen(Qt::red);

painter.drawPoint(r,height/2);

painter.drawPoint(s,height/2);

s+=25;

k++;

}\*/

}

void refl::on\_Draw\_clicked()

{

flag=0;

n=ui->textEdit->toPlainText().toInt();

update();

}

void refl::on\_xrefl\_clicked()

{

flag=1;

update();

}

void refl::on\_yrefl\_clicked()

{

flag=2;

update();

}

void refl::on\_xyrefl\_clicked()

{

flag=3;

update();

}

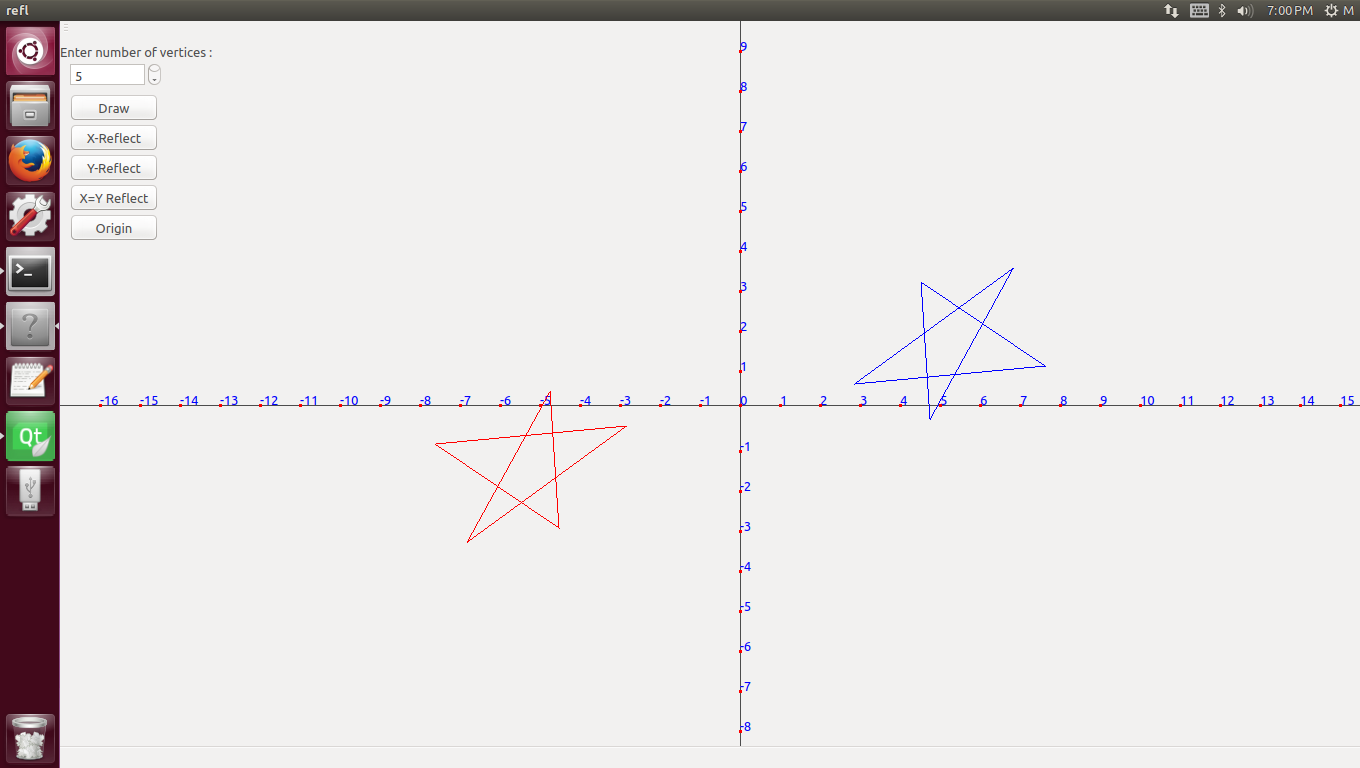
void refl::on\_Origin\_clicked()

{

flag=4;

update();

}

**OUTPUT: reflection on origin :**