Lab Report

For

Computer Graphics

Implementation Using Java

**Submitted By: Submitted To:**

Shahil Jha Mr. Manoj Giri

5th Semester, BIM Computer Graphics Instructor

Acknowledgement

This report has been written in order to fulfill the partial requirement of Tribhuvan University (TU) for the completion of the 5th semester of Bachelor of Information Management (BIM). Although this is an individual project assigned during the academic study in BIM Fifth Semester, I remain thankful to many people for the successful completion of this report.

First of all, I would like to thank my respected Computer Graphics Instructor, Manoj Giri, for his persistence in providing guidance throughout this academic study until its completion. His suggestions and guidance in every stage was a major factor in the completion of this report. Without his proper guidance, this report would not have been completed.

I am also thankful to my friends for providing aid when in confusion in the process of this report throughout this report.

At last, I would like to extend my thanks to our respected principle, Dr. Anil Lal Amatya for helping and encouraging in every aspect of our academic study and much appreciation to Asian School of Management and Technology, Kathmandu, Nepal.

Contents

[1. Translation 1](#_Toc33299093)

[2. Scaling 3](#_Toc33299094)

[a. About Origin 3](#_Toc33299095)

[b. About an Arbitrary Point 6](#_Toc33299096)

[3. Rotation 9](#_Toc33299097)

[a. About Origin 9](#_Toc33299098)

[b. About an Arbitrary Point 11](#_Toc33299099)

[4. Reflection 14](#_Toc33299100)

[a. About X-Axis 14](#_Toc33299101)

[b. About Y-Axis 18](#_Toc33299102)

[c. About Y=X 21](#_Toc33299103)

[d. About Y=-X 24](#_Toc33299104)

[e. About Origin 27](#_Toc33299105)

[f. About an Arbitrary Line 30](#_Toc33299106)

[5. Shear 33](#_Toc33299107)

[a. In X Direction 33](#_Toc33299108)

[b. In Y Direction 36](#_Toc33299109)

[c. In XY Direction 39](#_Toc33299110)

[6. Drawing Lines 41](#_Toc33299111)

[7. Drawing Strings 42](#_Toc33299112)

[8. Bresenham's Line Algorithm(BLA) 44](#_Toc33299113)

[9. Digital Differential Analyzer (DDA) Algorithm 46](#_Toc33299114)

[10. General Circle Drawing Algorithm 47](#_Toc33299115)

[a. About Origin 47](#_Toc33299116)

[b. At an Arbitrary Point 49](#_Toc33299117)

[11. Mid-Point Circle Drawing Algorithm 51](#_Toc33299118)

[a. About Origin 51](#_Toc33299119)

[b. About an Arbitrary Point 53](#_Toc33299120)

[12. Boundary Fill Algorithm 55](#_Toc33299121)

[13. Flood Fill Algorithm 57](#_Toc33299122)

# Translation

* Shahil\_Translation.java

**import** javax.swing.JFrame;

**public** **class** Shahil\_Translation{

**static** **int** *arr*[][] = {{100, 100},{150, 200}};

**static** **int**[][] *result* = **new** **int**[2][2];

**static** **int**[] *translate\_factor* = {200, 100};

**final** **static** **int** ***DEFAULT\_HEIGHT*** = 800;

**final** **static** **int** ***DEFAULT\_WIDTH*** = 800;

**final** **static** **int** ***DEFAULT\_X\_MID*** = ***DEFAULT\_HEIGHT***/2;

**final** **static** **int** ***DEFAULT\_Y\_MID*** = ***DEFAULT\_WIDTH***/2;

**public** **static** JFrame *frame*;

**public** Shahil\_Translation() {

//Frame making

*frame* = **new** JFrame();

*frame*.setTitle("Translation");

*frame*.setSize(***DEFAULT\_WIDTH***,***DEFAULT\_HEIGHT***);

*frame*.setVisible(**true**);

*frame*.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

}

**public** **void** Shahil\_drawQuartine(){

//Drawing of axes

*frame*.add(**new** Shahil\_Translation\_drawComponent(***DEFAULT\_HEIGHT***, ***DEFAULT\_WIDTH***, *arr*, *result*));

}

**public** **void** Shahil\_Opertaion() {

**for**(**int** i = 0 ; i<2; i++) {

*result*[i][0] = *arr*[i][0] + *translate\_factor*[0];

*result*[i][1] = *arr*[i][1] + *translate\_factor*[1];

}

}

**public** **static** **void** main(String[] args) {

Shahil\_Translation Shahil\_obj = **new** Shahil\_Translation();

Shahil\_obj.Shahil\_Opertaion();

Shahil\_obj.Shahil\_drawQuartine();

}

}

* Shahil\_Translation\_drawComponent.java

**import** java.awt.Color;

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** java.awt.geom.Line2D;

**import** java.awt.geom.Point2D;

**import** javax.swing.JComponent;

**public** **class** Shahil\_Translation\_drawComponent **extends** JComponent{

**static** **int** *DEFAULT\_HEIGHT*;

**static** **int** *DEFAULT\_WIDTH*;

**static** **int** *DEFAULT\_HEIGHT\_MID*;

**static** **int** *DEFAULT\_WIDTH\_MID*;

**static** **int** *arr*[][];

**static** **int** *result*[][];

**static** **int** *x1*,*y1*,*x2*,*y2*;

**static** **int** *a1*,*b1*,*a2*,*b2*;

Shahil\_Translation\_drawComponent(**int** DEFAULT\_HEIGHT, **int** DEFAULT\_WIDTH, **int**[][] arr, **int**[][] result){

**this**.*DEFAULT\_HEIGHT* = DEFAULT\_HEIGHT;

**this**.*DEFAULT\_WIDTH* = DEFAULT\_WIDTH;

*DEFAULT\_HEIGHT\_MID* = DEFAULT\_HEIGHT/2;

*DEFAULT\_WIDTH\_MID* = DEFAULT\_WIDTH/2;

**this**.*arr* = arr;

**this**.*result* = result;

*Shahil\_mapArray*();

}

**public** **static** **void** Shahil\_mapArray() {

*x1* = *arr*[0][0];

*y1* = *arr*[0][1];

*x2* = *arr* [1][0];

*y2* = *arr*[1][1];

*a1* = *result*[0][0];

*b1* = *result*[0][1];

*a2* = *result* [1][0];

*b2* = *result*[1][1];

}

**public** **void** paintComponent(Graphics g){

Graphics2D g2d = (Graphics2D) g;

//Making the x and y axes

Line2D.Double x\_line = **new** Line2D.Double(**new** Point2D.Double(0,*DEFAULT\_HEIGHT\_MID*), **new** Point2D.Double(*DEFAULT\_WIDTH*,*DEFAULT\_HEIGHT\_MID*));

Line2D.Double y\_line = **new** Line2D.Double(**new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, 0), **new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, *DEFAULT\_HEIGHT*));

g2d.draw(x\_line);

g2d.draw(y\_line);

g2d.rotate(Math.*toRadians*(-90),400,400);

g2d.translate(400, 400);

Line2D.Double Line1 = **new** Line2D.Double(**new** Point2D.Double(*x1*, *y1*), **new** Point2D.Double(*x2*,*y2*));

g2d.draw(Line1);

g2d.setColor(Color.***RED***);

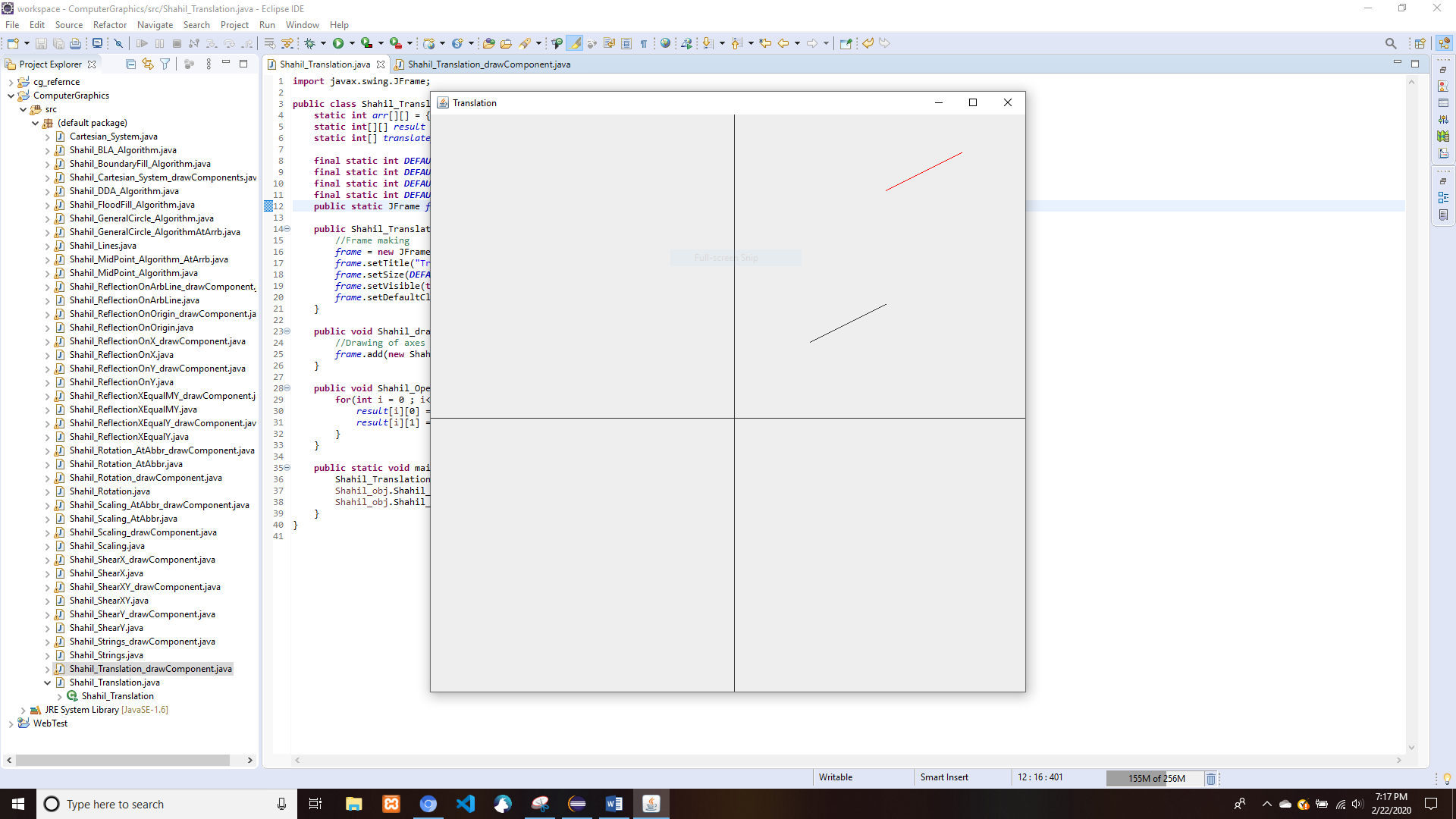
Line2D.Double Line2 = **new** Line2D.Double(**new** Point2D.Double(*a1*,*b1*), **new** Point2D.Double(*a2*,*b2*));

g2d.draw(Line2);

}

}

* Output



# Scaling

## About Origin

* Shahil\_Scaling.java

**import** javax.swing.JFrame;

**public** **class** Shahil\_Scaling {

**static** **int** *arr\_x*[] = {123,100,156};

**static** **int** *arr\_y*[] = {90,167,78};

**static** **int**[] *res\_x* = **new** **int**[3];

**static** **int**[] *res\_y* = **new** **int**[3];

**static** **int**[] *scaling\_factor* = {2, 2};

**final** **static** **int** ***DEFAULT\_HEIGHT*** = 800;

**final** **static** **int** ***DEFAULT\_WIDTH*** = 800;

**final** **static** **int** ***DEFAULT\_X\_MID*** = ***DEFAULT\_HEIGHT***/2;

**final** **static** **int** ***DEFAULT\_Y\_MID*** = ***DEFAULT\_WIDTH***/2;

**public** **static** JFrame *frame*;

**public** Shahil\_Scaling() {

//Frame making

*frame* = **new** JFrame();

*frame*.setTitle("Scaling");

*frame*.setSize(***DEFAULT\_WIDTH***,***DEFAULT\_HEIGHT***);

*frame*.setVisible(**true**);

*frame*.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

}

**public** **void** Shahil\_drawQuartine(){

//Drawing of axes

*frame*.add(**new** Shahil\_Scaling\_drawComponent(***DEFAULT\_HEIGHT***, ***DEFAULT\_WIDTH***, *arr\_x*, *arr\_y* , *res\_x*, *res\_y*));

}

**public** **void** Shahil\_Opertaion() {

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

*res\_x*[i] = *arr\_x*[i] \* *scaling\_factor*[0];

*res\_y*[i] = *arr\_y*[i] \* *scaling\_factor*[1];

System.***out***.println(*res\_x*[i]+"\t"+*res\_y*[i]);

}

}

**public** **static** **void** main(String[] args) {

Shahil\_Scaling Shahil\_obj = **new** Shahil\_Scaling();

Shahil\_obj.Shahil\_Opertaion();

Shahil\_obj.Shahil\_drawQuartine();

}

}

* Shahil\_Scaling\_drawComponent.java

**import** java.awt.Color;

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** java.awt.Polygon;

**import** java.awt.Rectangle;

**import** java.awt.geom.Line2D;

**import** java.awt.geom.Point2D;

**import** javax.swing.JComponent;

**public** **class** Shahil\_Scaling\_drawComponent **extends** JComponent{

**static** **int** *DEFAULT\_HEIGHT*;

**static** **int** *DEFAULT\_WIDTH*;

**static** **int** *DEFAULT\_HEIGHT\_MID*;

**static** **int** *DEFAULT\_WIDTH\_MID*;

**static** **int** *arr\_x*[];

**static** **int** *arr\_y*[];

**static** **int** *res\_x*[];

**static** **int** *res\_y*[];

**static** **int** *x1*, *x2*, *x3*;

**static** **int** *y1*, *y2*, *y3*;

Shahil\_Scaling\_drawComponent(**int** DEFAULT\_HEIGHT, **int** DEFAULT\_WIDTH, **int**[] arr\_x, **int**[] arr\_y, **int**[] res\_x, **int**[] res\_y){

**this**.*DEFAULT\_HEIGHT* = DEFAULT\_HEIGHT;

**this**.*DEFAULT\_WIDTH* = DEFAULT\_WIDTH;

*DEFAULT\_HEIGHT\_MID* = DEFAULT\_HEIGHT/2;

*DEFAULT\_WIDTH\_MID* = DEFAULT\_WIDTH/2;

**this**.*arr\_x* = arr\_x;

**this**.*arr\_y* = arr\_y;

**this**.*res\_x* = res\_x;

**this**.*res\_y* = res\_y;

}

**public** **void** paintComponent(Graphics g){

Graphics2D g2d = (Graphics2D) g;

//Making the x and y axes

Line2D.Double x\_line = **new** Line2D.Double(**new** Point2D.Double(0,*DEFAULT\_HEIGHT\_MID*), **new** Point2D.Double(*DEFAULT\_WIDTH*,*DEFAULT\_HEIGHT\_MID*));

Line2D.Double y\_line = **new** Line2D.Double(**new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, 0), **new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, *DEFAULT\_HEIGHT*));

g2d.draw(x\_line);

g2d.draw(y\_line);

g2d.rotate(Math.*toRadians*(-90),400,400);

g2d.translate(400, 400);

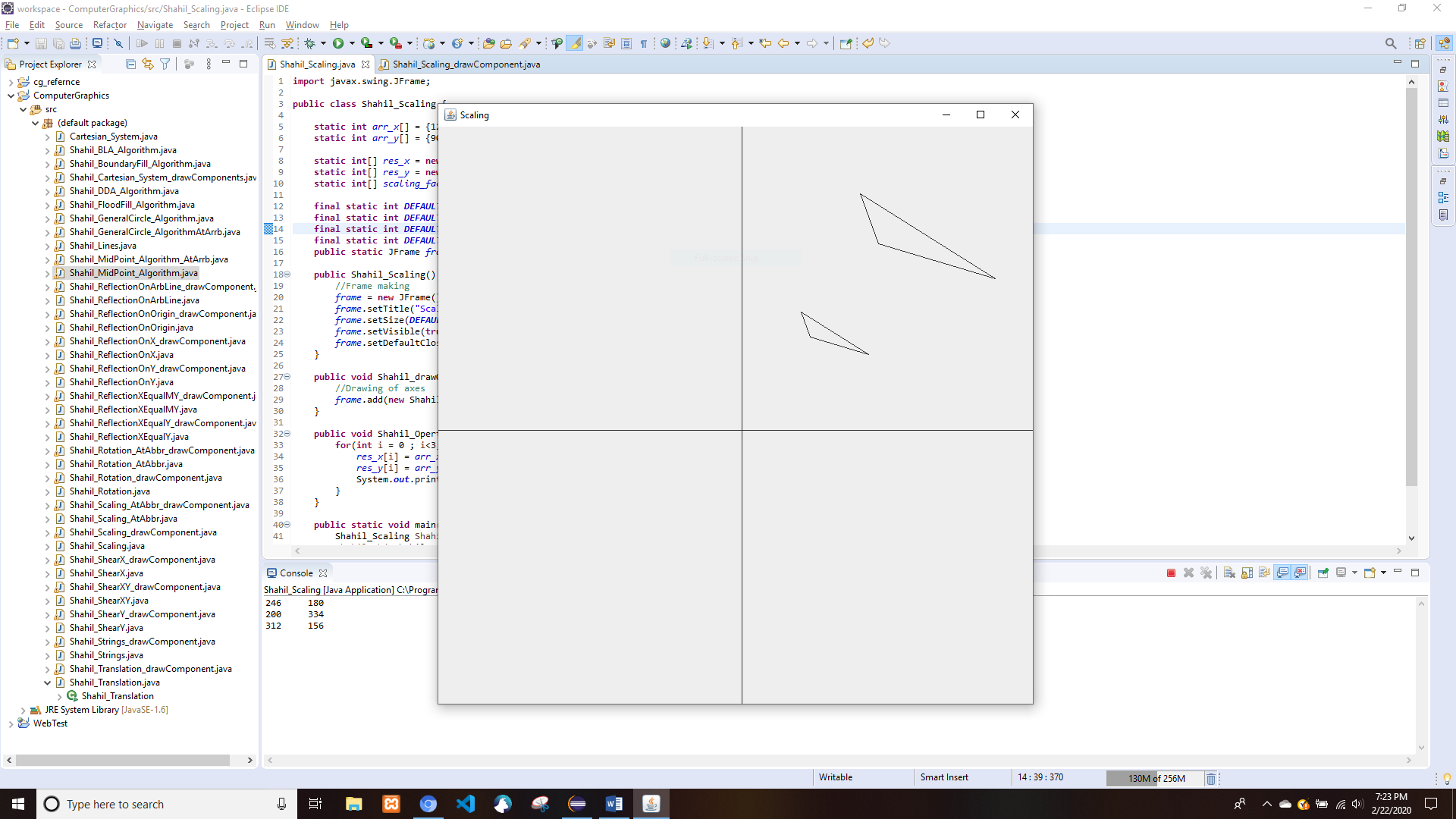
g2d.drawPolygon(*arr\_x*, *arr\_y*, *arr\_x*.length);

g2d.drawPolygon(*res\_x*, *res\_y*, *res\_x*.length);

}

}

* Output



## About an Arbitrary Point

* Shahil\_Scaling\_AtAbbr.java

**import** javax.swing.JFrame;

**public** **class** Shahil\_Scaling\_AtAbbr {

**static** **int** *arr\_x*[] = {123,100,156};

**static** **int** *arr\_y*[] = {90,167,78};

**static** **int**[] *res\_x* = **new** **int**[3];

**static** **int**[] *res\_y* = **new** **int**[3];

**static** **int**[] *scaling\_factor* = {2, 2};

**final** **static** **int** ***DEFAULT\_HEIGHT*** = 800;

**final** **static** **int** ***DEFAULT\_WIDTH*** = 800;

**final** **static** **int** ***DEFAULT\_X\_MID*** = ***DEFAULT\_HEIGHT***/2;

**final** **static** **int** ***DEFAULT\_Y\_MID*** = ***DEFAULT\_WIDTH***/2;

**int** abbr\_x = 50;//Arbitrary point x

**int** abbr\_y = 50;//Arbitrary point y

**public** **static** JFrame *frame*;

**public** Shahil\_Scaling\_AtAbbr() {

//Frame making

*frame* = **new** JFrame();

*frame*.setTitle("Scaling");

*frame*.setSize(***DEFAULT\_WIDTH***,***DEFAULT\_HEIGHT***);

*frame*.setVisible(**true**);

*frame*.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

}

**public** **void** Shahil\_drawQuartine(){

//Drawing of axes

*frame*.add(**new** Shahil\_Scaling\_drawComponent(***DEFAULT\_HEIGHT***, ***DEFAULT\_WIDTH***, *arr\_x*, *arr\_y* , *res\_x*, *res\_y*));

}

**public** **static** **void** Shahil\_Translate(**int** h, **int** k) {

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

*res\_x*[i] = *arr\_x*[i] + h;

*res\_y*[i] = *arr\_y*[i] + k;

}

}

**public** **void** Shahil\_Opertaion() {

*Shahil\_Translate*(-abbr\_x, -abbr\_y);

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

*res\_x*[i] = *arr\_x*[i] \* *scaling\_factor*[0];

*res\_y*[i] = *arr\_y*[i] \* *scaling\_factor*[1];

System.***out***.println(*res\_x*[i]+"\t"+*res\_y*[i]);

}

*Shahil\_Translate*(abbr\_x, abbr\_y);

}

**public** **static** **void** main(String[] args) {

Shahil\_Scaling\_AtAbbr Shahil\_obj = **new** Shahil\_Scaling\_AtAbbr();

Shahil\_obj.Shahil\_Opertaion();

Shahil\_obj.Shahil\_drawQuartine();

}

}

* Shahil\_Scaling\_AtAbbr\_drawComponent.java

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** java.awt.geom.Line2D;

**import** java.awt.geom.Point2D;

**import** javax.swing.JComponent;

**public** **class** Shahil\_Scaling\_AtAbbr\_drawComponent **extends** JComponent {

**static** **int** *DEFAULT\_HEIGHT*;

**static** **int** *DEFAULT\_WIDTH*;

**static** **int** *DEFAULT\_HEIGHT\_MID*;

**static** **int** *DEFAULT\_WIDTH\_MID*;

**static** **int** *arr\_x*[];

**static** **int** *arr\_y*[];

**static** **int** *res\_x*[];

**static** **int** *res\_y*[];

**static** **int** *x1*, *x2*, *x3*;

**static** **int** *y1*, *y2*, *y3*;

Shahil\_Scaling\_AtAbbr\_drawComponent(**int** DEFAULT\_HEIGHT, **int** DEFAULT\_WIDTH, **int**[] arr\_x, **int**[] arr\_y, **int**[] res\_x, **int**[] res\_y){

**this**.*DEFAULT\_HEIGHT* = DEFAULT\_HEIGHT;

**this**.*DEFAULT\_WIDTH* = DEFAULT\_WIDTH;

*DEFAULT\_HEIGHT\_MID* = DEFAULT\_HEIGHT/2;

*DEFAULT\_WIDTH\_MID* = DEFAULT\_WIDTH/2;

**this**.*arr\_x* = arr\_x;

**this**.*arr\_y* = arr\_y;

**this**.*res\_x* = res\_x;

**this**.*res\_y* = res\_y;

}

**public** **void** paintComponent(Graphics g){

Graphics2D g2d = (Graphics2D) g;

//Making the x and y axes

Line2D.Double x\_line = **new** Line2D.Double(**new** Point2D.Double(0,*DEFAULT\_HEIGHT\_MID*), **new** Point2D.Double(*DEFAULT\_WIDTH*,*DEFAULT\_HEIGHT\_MID*));

Line2D.Double y\_line = **new** Line2D.Double(**new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, 0), **new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, *DEFAULT\_HEIGHT*));

g2d.draw(x\_line);

g2d.draw(y\_line);

g2d.rotate(Math.*toRadians*(-90),400,400);

g2d.translate(400, 400);

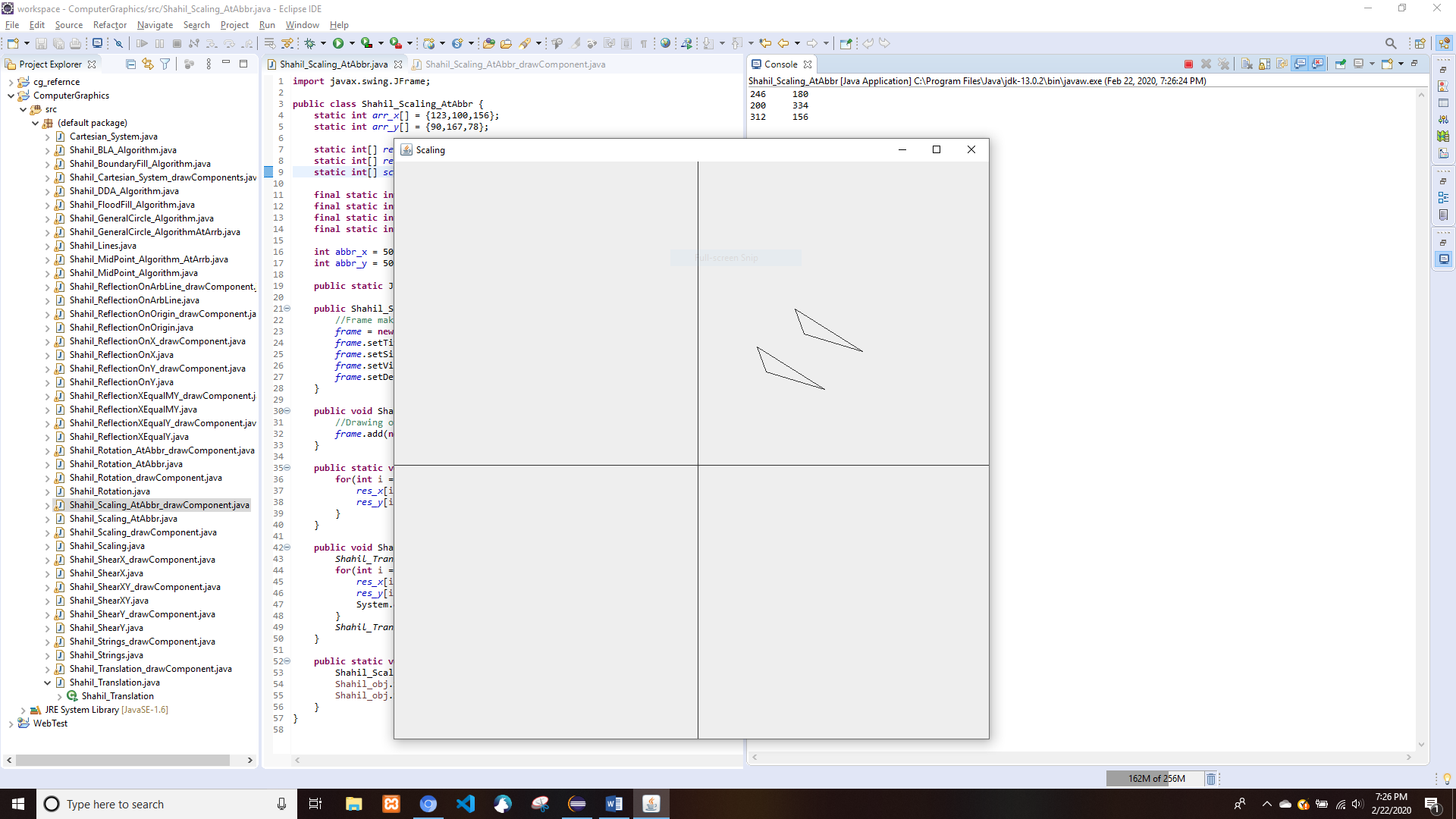
g2d.drawPolygon(*arr\_x*, *arr\_y*, *arr\_x*.length);

g2d.drawPolygon(*res\_x*, *res\_y*, *res\_x*.length);

}

}

* Output



# Rotation

## About Origin

* Shahil\_Rotation.java

**import** javax.swing.JFrame;

**public** **class** Shahil\_Rotation {

**static** **int** *arr*[][] = {

{100, 100},

{150, 200}

};

**static** **int**[][] *result* = **new** **int**[2][2];

**final** **static** **int** ***DEFAULT\_HEIGHT*** = 800;

**final** **static** **int** ***DEFAULT\_WIDTH*** = 800;

**final** **static** **int** ***DEFAULT\_X\_MID*** = ***DEFAULT\_HEIGHT***/2;

**final** **static** **int** ***DEFAULT\_Y\_MID*** = ***DEFAULT\_WIDTH***/2;

**public** **static** JFrame *frame*;

**public** Shahil\_Rotation() {

//Frame making

*frame* = **new** JFrame();

*frame*.setTitle("Anti Clockwise Rotatation");

*frame*.setSize(***DEFAULT\_WIDTH***,***DEFAULT\_HEIGHT***);

*frame*.setVisible(**true**);

*frame*.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

}

**public** **void** Shahil\_drawQuartine(){

//Drawing of axes

*frame*.add(**new** Shahil\_Rotation\_drawComponent(***DEFAULT\_HEIGHT***, ***DEFAULT\_WIDTH***, *arr*, *result*));

}

**public** **void** Shahil\_Opertaion() {

**double** angleRadian= Math.*toRadians*(180);

**for**(**int** i = 0 ; i<2; i++) {

*result*[i][0]=(**int**)Math.*round*(*arr*[i][0]\*Math.*cos*(angleRadian)+*arr*[i][1]\*Math.*sin*(angleRadian));

*result*[i][1]=(**int**)Math.*round*(*arr*[i][0]\*Math.*sin*(angleRadian)+*arr*[i][1]\*Math.*cos*(angleRadian));

}

}

**public** **static** **void** main(String[] args) {

Shahil\_Rotation Shahil\_obj = **new** Shahil\_Rotation();

Shahil\_obj.Shahil\_Opertaion();

Shahil\_obj.Shahil\_drawQuartine();

}

}

* Shahil\_Rotation\_drawComponent.java

**import** java.awt.Color;

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** java.awt.geom.Line2D;

**import** java.awt.geom.Point2D;

**import** javax.swing.JComponent;

**public** **class** Shahil\_Rotation\_drawComponent **extends** JComponent {

**static** **int** *DEFAULT\_HEIGHT*;

**static** **int** *DEFAULT\_WIDTH*;

**static** **int** *DEFAULT\_HEIGHT\_MID*;

**static** **int** *DEFAULT\_WIDTH\_MID*;

**static** **int** *arr*[][];

**static** **int** *result*[][];

**static** **int** *x1*,*y1*,*x2*,*y2*;

**static** **int** *a1*,*b1*,*a2*,*b2*;

Shahil\_Rotation\_drawComponent(**int** DEFAULT\_HEIGHT, **int** DEFAULT\_WIDTH, **int**[][] arr, **int**[][] result){

**this**.*DEFAULT\_HEIGHT* = DEFAULT\_HEIGHT;

**this**.*DEFAULT\_WIDTH* = DEFAULT\_WIDTH;

*DEFAULT\_HEIGHT\_MID* = DEFAULT\_HEIGHT/2;

*DEFAULT\_WIDTH\_MID* = DEFAULT\_WIDTH/2;

**this**.*arr* = arr;

**this**.*result* = result;

*Shahil\_mapArray*();

}

**public** **static** **void** Shahil\_mapArray() {

*x1* = *arr*[0][0];

*y1* = *arr*[0][1];

*x2* = *arr* [1][0];

*y2* = *arr*[1][1];

*a1* = *result*[0][0];

*b1* = *result*[0][1];

*a2* = *result* [1][0];

*b2* = *result*[1][1];

}

**public** **void** paintComponent(Graphics g){

Graphics2D g2d = (Graphics2D) g;

//Making the x and y axes

Line2D.Double x\_line = **new** Line2D.Double(**new** Point2D.Double(0,*DEFAULT\_HEIGHT\_MID*), **new** Point2D.Double(*DEFAULT\_WIDTH*,*DEFAULT\_HEIGHT\_MID*));

Line2D.Double y\_line = **new** Line2D.Double(**new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, 0), **new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, *DEFAULT\_HEIGHT*));

g2d.draw(x\_line);

g2d.draw(y\_line);

g2d.rotate(Math.*toRadians*(-90),400,400);

g2d.translate(400, 400);

Line2D.Double Line1 = **new** Line2D.Double(**new** Point2D.Double(*x1*, *y1*), **new** Point2D.Double(*x2*,*y2*));

g2d.draw(Line1);

g2d.setColor(Color.***RED***);

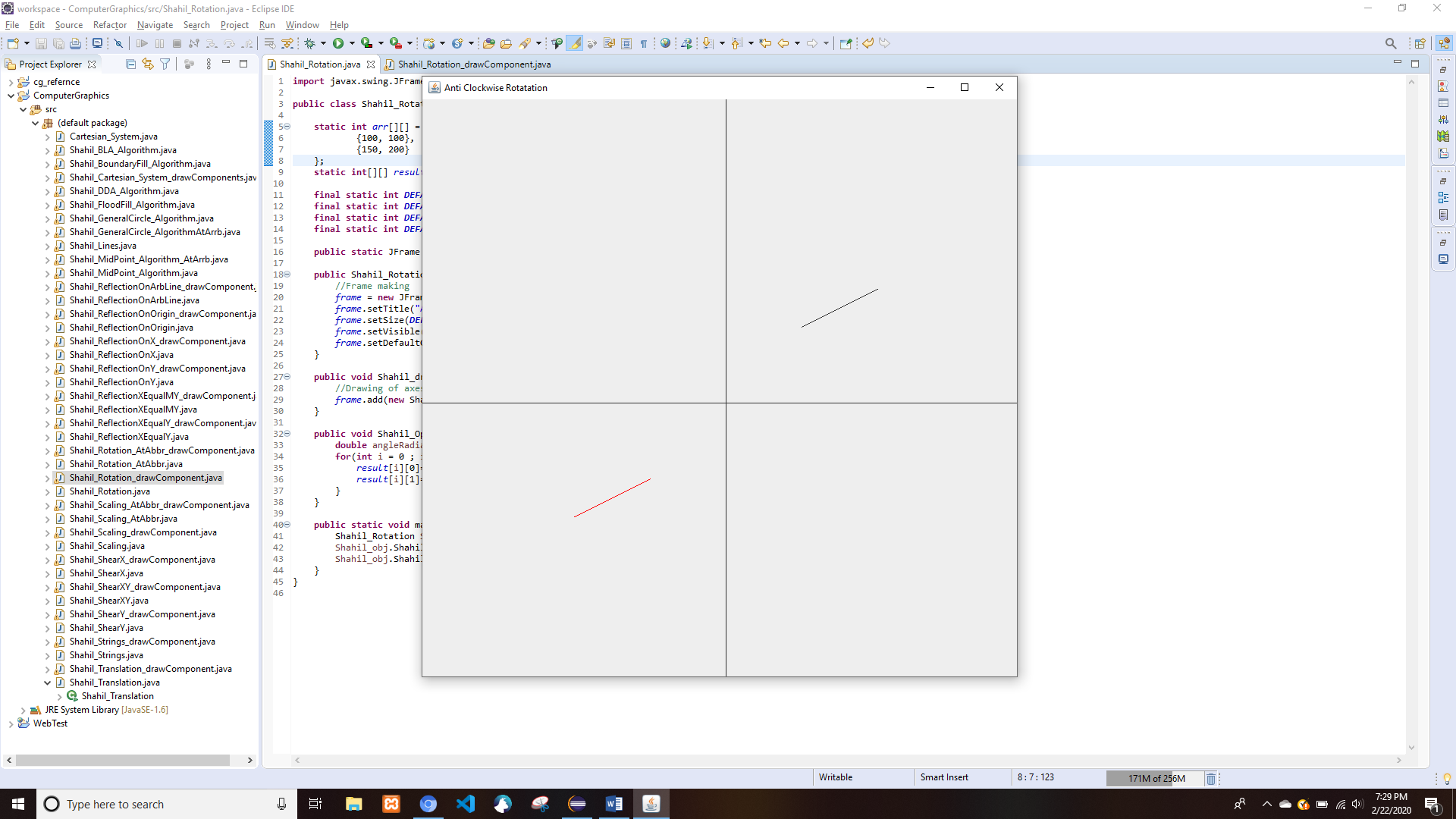
Line2D.Double Line2 = **new** Line2D.Double(**new** Point2D.Double(*a1*,*b1*), **new** Point2D.Double(*a2*,*b2*));

g2d.draw(Line2);

}

}

* Output



## About an Arbitrary Point

* Shahil\_Rotation\_AtAbbr.java

**import** javax.swing.JFrame;

**public** **class** Shahil\_Rotation\_AtAbbr {

**static** **int** *arr*[][] = {

{100, 100},

{150, 200}

};

**static** **int**[][] *result* = **new** **int**[2][2];

**final** **static** **int** ***DEFAULT\_HEIGHT*** = 800;

**final** **static** **int** ***DEFAULT\_WIDTH*** = 800;

**final** **static** **int** ***DEFAULT\_X\_MID*** = ***DEFAULT\_HEIGHT***/2;

**final** **static** **int** ***DEFAULT\_Y\_MID*** = ***DEFAULT\_WIDTH***/2;

**int** abbr\_x = 50;//Arbitrary point x

**int** abbr\_y = 50;//Arbitrary point y

**public** **static** JFrame *frame*;

**public** Shahil\_Rotation\_AtAbbr() {

//Frame making

*frame* = **new** JFrame();

*frame*.setTitle("Anti Clockwise Rotatation");

*frame*.setSize(***DEFAULT\_WIDTH***,***DEFAULT\_HEIGHT***);

*frame*.setVisible(**true**);

*frame*.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

}

**public** **void** Shahil\_drawQuartine(){

//Drawing of axes

*frame*.add(**new** Shahil\_Rotation\_AtAbbr\_drawComponent(***DEFAULT\_HEIGHT***, ***DEFAULT\_WIDTH***, *arr*, *result*));

}

**public** **static** **void** Shahil\_Translate(**int** h, **int** k) {

**for**(**int** i = 0 ; i<2; i++) {

*result*[i][0] = *arr*[i][0] + h;

*result*[i][1] = *arr*[i][1] + k;

System.***out***.println("Translated \t"+*result*[i][0]+"\t"+*result*[i][1]);

}

}

**public** **void** Shahil\_Opertaion() {

*Shahil\_Translate*(-abbr\_x,-abbr\_y);

**double** angleRadian= Math.*toRadians*(180);

**for**(**int** i = 0 ; i<2; i++) {

*result*[i][0]=(**int**)Math.*round*(*arr*[i][0]\*Math.*cos*(angleRadian)+*arr*[i][1]\*Math.*sin*(angleRadian));

*result*[i][1]=(**int**)Math.*round*(*arr*[i][0]\*Math.*sin*(angleRadian)+*arr*[i][1]\*Math.*cos*(angleRadian));

System.***out***.println("rotated \t"+*result*[i][0]+"\t"+*result*[i][1]);

}

*Shahil\_Translate*(abbr\_x,abbr\_y);

}

**public** **static** **void** main(String[] args) {

Shahil\_Rotation\_AtAbbr Shahil\_obj = **new** Shahil\_Rotation\_AtAbbr();

Shahil\_obj.Shahil\_Opertaion();

Shahil\_obj.Shahil\_drawQuartine();

}

}

* Shahil\_Rotation\_AtAbbr\_drawComponent.java

**import** java.awt.Color;

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** java.awt.geom.Line2D;

**import** java.awt.geom.Point2D;

**import** javax.swing.JComponent;

**public** **class** Shahil\_Rotation\_AtAbbr\_drawComponent **extends** JComponent{

**static** **int** *DEFAULT\_HEIGHT*;

**static** **int** *DEFAULT\_WIDTH*;

**static** **int** *DEFAULT\_HEIGHT\_MID*;

**static** **int** *DEFAULT\_WIDTH\_MID*;

**static** **int** *arr*[][];

**static** **int** *result*[][];

**static** **int** *x1*,*y1*,*x2*,*y2*;

**static** **int** *a1*,*b1*,*a2*,*b2*;

Shahil\_Rotation\_AtAbbr\_drawComponent(**int** DEFAULT\_HEIGHT, **int** DEFAULT\_WIDTH, **int**[][] arr, **int**[][] result){

**this**.*DEFAULT\_HEIGHT* = DEFAULT\_HEIGHT;

**this**.*DEFAULT\_WIDTH* = DEFAULT\_WIDTH;

*DEFAULT\_HEIGHT\_MID* = DEFAULT\_HEIGHT/2;

*DEFAULT\_WIDTH\_MID* = DEFAULT\_WIDTH/2;

**this**.*arr* = arr;

**this**.*result* = result;

*Shahil\_mapArray*();

}

**public** **static** **void** Shahil\_mapArray() {

*x1* = *arr*[0][0];

*y1* = *arr*[0][1];

*x2* = *arr* [1][0];

*y2* = *arr*[1][1];

*a1* = *result*[0][0];

*b1* = *result*[0][1];

*a2* = *result* [1][0];

*b2* = *result*[1][1];

}

**public** **void** paintComponent(Graphics g){

Graphics2D g2d = (Graphics2D) g;

//Making the x and y axes

Line2D.Double x\_line = **new** Line2D.Double(**new** Point2D.Double(0,*DEFAULT\_HEIGHT\_MID*), **new** Point2D.Double(*DEFAULT\_WIDTH*,*DEFAULT\_HEIGHT\_MID*));

Line2D.Double y\_line = **new** Line2D.Double(**new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, 0), **new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, *DEFAULT\_HEIGHT*));

g2d.draw(x\_line);

g2d.draw(y\_line);

g2d.rotate(Math.*toRadians*(-90),400,400);

g2d.translate(400, 400);

Line2D.Double Line1 = **new** Line2D.Double(**new** Point2D.Double(*x1*, *y1*), **new** Point2D.Double(*x2*,*y2*));

g2d.draw(Line1);

g2d.setColor(Color.***RED***);

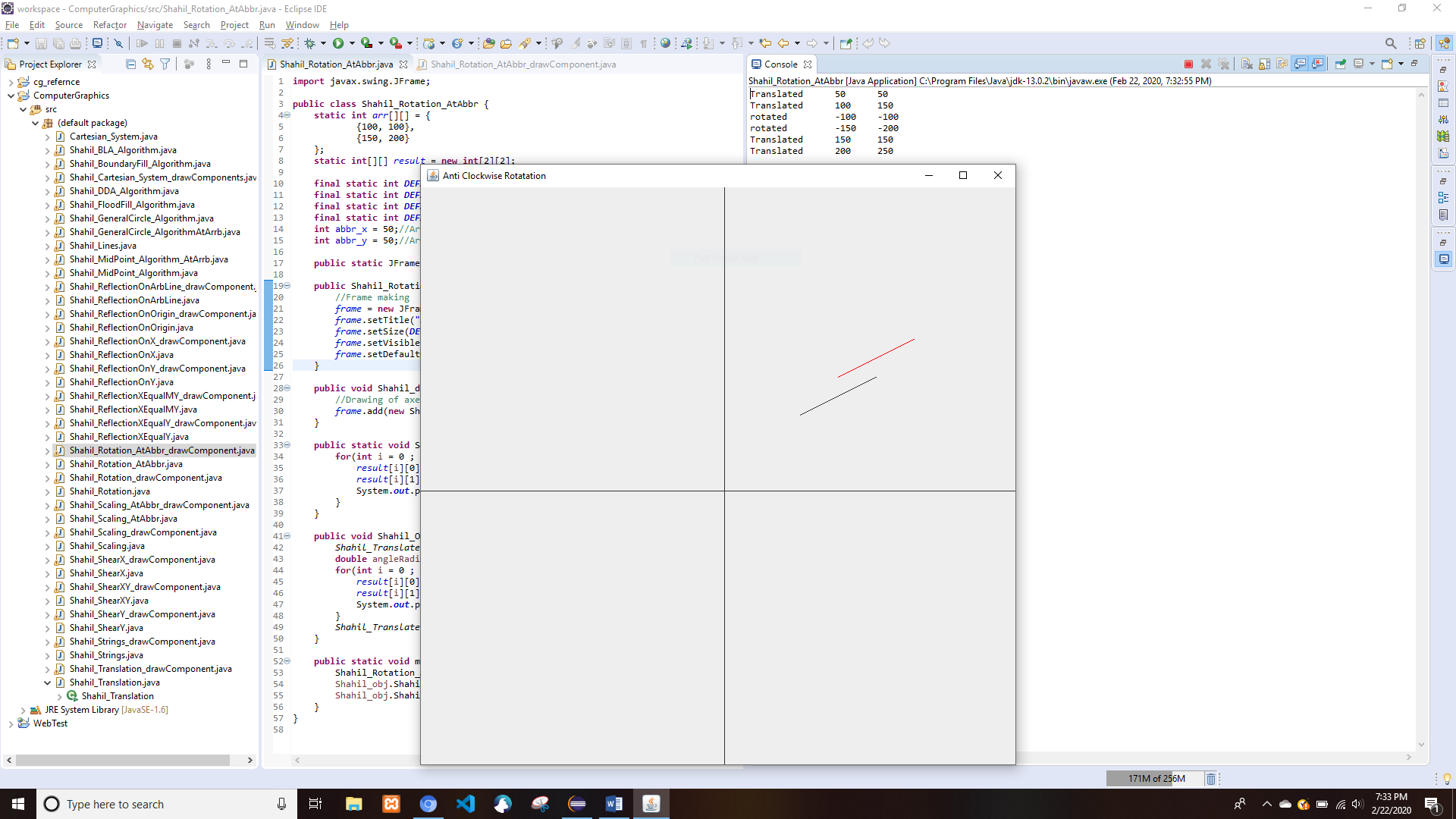
Line2D.Double Line2 = **new** Line2D.Double(**new** Point2D.Double(*a1*,*b1*), **new** Point2D.Double(*a2*,*b2*));

g2d.draw(Line2);

}

}

* Output



# Reflection

## About X-Axis

* Shahil\_ReflectionOnX.java

**import** javax.swing.JFrame;

**public** **class** Shahil\_ReflectionOnX {

**static** **int** *arr*[][] = {

{100, 100},

{150, 200}

};

**static** **int**[][] *result* = **new** **int**[2][2];

**final** **static** **int** ***DEFAULT\_HEIGHT*** = 800;

**final** **static** **int** ***DEFAULT\_WIDTH*** = 800;

**final** **static** **int** ***DEFAULT\_X\_MID*** = ***DEFAULT\_HEIGHT***/2;

**final** **static** **int** ***DEFAULT\_Y\_MID*** = ***DEFAULT\_WIDTH***/2;

**public** **static** JFrame *frame*;

**public** Shahil\_ReflectionOnX() {

//Frame making

*frame* = **new** JFrame();

*frame*.setTitle("Reflection On Origin");

*frame*.setSize(***DEFAULT\_WIDTH***,***DEFAULT\_HEIGHT***);

*frame*.setVisible(**true**);

*frame*.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

}

**public** **void** Shahil\_drawQuartine(){

//Drawing of axes

*frame*.add(**new** Shahil\_ReflectionOnX\_drawComponent(***DEFAULT\_HEIGHT***, ***DEFAULT\_WIDTH***, *arr*, *result*));

}

**public** **void** Shahil\_Opertaion() {

**for**(**int** i = 0 ; i<2; i++) {

*result*[i][0] = *arr*[i][0] \* -1;

*result*[i][1] = *arr*[i][1] \* 1;

System.***out***.println(*result*[i][0]+"\t"+*result*[i][1]);

}

}

**public** **static** **void** main(String[] args) {

Shahil\_ReflectionOnX Shahil\_obj = **new** Shahil\_ReflectionOnX();

Shahil\_obj.Shahil\_Opertaion();

Shahil\_obj.Shahil\_drawQuartine();

}

}

* Shahil\_ReflectionOnX\_drawComponent.java

**import** java.awt.Color;

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** java.awt.geom.Line2D;

**import** java.awt.geom.Point2D;

**import** javax.swing.JComponent;

**public** **class** Shahil\_ReflectionOnX\_drawComponent **extends** JComponent {

**static** **int** *DEFAULT\_HEIGHT*;

**static** **int** *DEFAULT\_WIDTH*;

**static** **int** *DEFAULT\_HEIGHT\_MID*;

**static** **int** *DEFAULT\_WIDTH\_MID*;

**static** **int** *arr*[][];

**static** **int** *result*[][];

**static** **int** *x1*,*y1*,*x2*,*y2*;

**static** **int** *a1*,*b1*,*a2*,*b2*;

Shahil\_ReflectionOnX\_drawComponent(**int** DEFAULT\_HEIGHT, **int** DEFAULT\_WIDTH, **int**[][] arr, **int**[][] result){

**this**.*DEFAULT\_HEIGHT* = DEFAULT\_HEIGHT;

**this**.*DEFAULT\_WIDTH* = DEFAULT\_WIDTH;

*DEFAULT\_HEIGHT\_MID* = DEFAULT\_HEIGHT/2;

*DEFAULT\_WIDTH\_MID* = DEFAULT\_WIDTH/2;

**this**.*arr* = arr;

**this**.*result* = result;

*Shahil\_mapArray*();

}

**public** **static** **void** Shahil\_mapArray() {

*x1* = *arr*[0][0];

*y1* = *arr*[0][1];

*x2* = *arr* [1][0];

*y2* = *arr*[1][1];

*a1* = *result*[0][0];

*b1* = *result*[0][1];

*a2* = *result* [1][0];

*b2* = *result*[1][1];

}

**public** **void** paintComponent(Graphics g){

Graphics2D g2d = (Graphics2D) g;

//Making the x and y axes

Line2D.Double x\_line = **new** Line2D.Double(**new** Point2D.Double(0,*DEFAULT\_HEIGHT\_MID*), **new** Point2D.Double(*DEFAULT\_WIDTH*,*DEFAULT\_HEIGHT\_MID*));

Line2D.Double y\_line = **new** Line2D.Double(**new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, 0), **new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, *DEFAULT\_HEIGHT*));

g2d.draw(x\_line);

g2d.draw(y\_line);

g2d.rotate(Math.*toRadians*(-90),400,400);

g2d.translate(400, 400);

Line2D.Double Line1 = **new** Line2D.Double(**new** Point2D.Double(*x1*, *y1*), **new** Point2D.Double(*x2*,*y2*));

g2d.draw(Line1);

g2d.setColor(Color.***RED***);

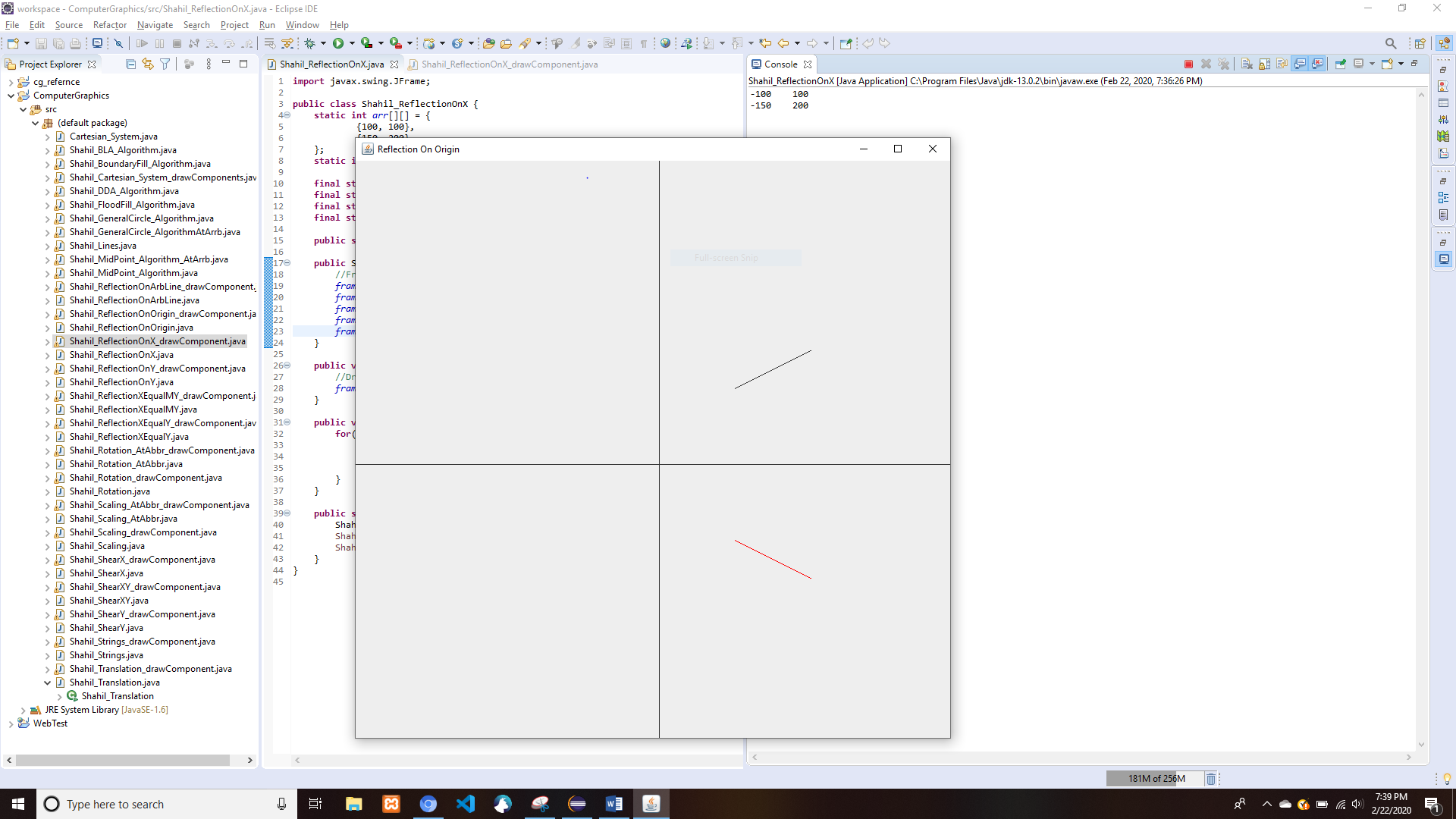
Line2D.Double Line2 = **new** Line2D.Double(**new** Point2D.Double(*a1*,*b1*), **new** Point2D.Double(*a2*,*b2*));

g2d.draw(Line2);

}

}

* Output



## About Y-Axis

* Shahil\_ReflectionOnY.java

**import** javax.swing.JFrame;

**public** **class** Shahil\_ReflectionOnY {

**static** **int** *arr*[][] = {

{100, 100},

{150, 200}

};

**static** **int**[][] *result* = **new** **int**[2][2];

**final** **static** **int** ***DEFAULT\_HEIGHT*** = 800;

**final** **static** **int** ***DEFAULT\_WIDTH*** = 800;

**final** **static** **int** ***DEFAULT\_X\_MID*** = ***DEFAULT\_HEIGHT***/2;

**final** **static** **int** ***DEFAULT\_Y\_MID*** = ***DEFAULT\_WIDTH***/2;

**public** **static** JFrame *frame*;

**public** Shahil\_ReflectionOnY() {

//Frame making

*frame* = **new** JFrame();

*frame*.setTitle("Reflection On Origin");

*frame*.setSize(***DEFAULT\_WIDTH***,***DEFAULT\_HEIGHT***);

*frame*.setVisible(**true**);

*frame*.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

}

**public** **void** Shahil\_drawQuartine(){

//Drawing of axes

*frame*.add(**new** Shahil\_ReflectionOnY\_drawComponent(***DEFAULT\_HEIGHT***, ***DEFAULT\_WIDTH***, *arr*, *result*));

}

**public** **void** Shahil\_Opertaion() {

**for**(**int** i = 0 ; i<2; i++) {

*result*[i][0] = *arr*[i][0] \* 1;

*result*[i][1] = *arr*[i][1] \* -1;

System.***out***.println(*result*[i][0]+"\t"+*result*[i][1]);

}

}

**public** **static** **void** main(String[] args) {

Shahil\_ReflectionOnY Shahil\_obj = **new** Shahil\_ReflectionOnY();

Shahil\_obj.Shahil\_Opertaion();

Shahil\_obj.Shahil\_drawQuartine();

}

}

* Shahil\_ReflectionOnY\_drawComponent.java

**import** java.awt.Color;

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** java.awt.geom.Line2D;

**import** java.awt.geom.Point2D;

**import** javax.swing.JComponent;

**public** **class** Shahil\_ReflectionOnY\_drawComponent **extends** JComponent {

**static** **int** *DEFAULT\_HEIGHT*;

**static** **int** *DEFAULT\_WIDTH*;

**static** **int** *DEFAULT\_HEIGHT\_MID*;

**static** **int** *DEFAULT\_WIDTH\_MID*;

**static** **int** *arr*[][];

**static** **int** *result*[][];

**static** **int** *x1*,*y1*,*x2*,*y2*;

**static** **int** *a1*,*b1*,*a2*,*b2*;

Shahil\_ReflectionOnY\_drawComponent(**int** DEFAULT\_HEIGHT, **int** DEFAULT\_WIDTH, **int**[][] arr, **int**[][] result){

**this**.*DEFAULT\_HEIGHT* = DEFAULT\_HEIGHT;

**this**.*DEFAULT\_WIDTH* = DEFAULT\_WIDTH;

*DEFAULT\_HEIGHT\_MID* = DEFAULT\_HEIGHT/2;

*DEFAULT\_WIDTH\_MID* = DEFAULT\_WIDTH/2;

**this**.*arr* = arr;

**this**.*result* = result;

*Shahil\_mapArray*();

}

**public** **static** **void** Shahil\_mapArray() {

*x1* = *arr*[0][0];

*y1* = *arr*[0][1];

*x2* = *arr* [1][0];

*y2* = *arr*[1][1];

*a1* = *result*[0][0];

*b1* = *result*[0][1];

*a2* = *result* [1][0];

*b2* = *result*[1][1];

}

**public** **void** paintComponent(Graphics g){

Graphics2D g2d = (Graphics2D) g;

//Making the x and y axes

Line2D.Double x\_line = **new** Line2D.Double(**new** Point2D.Double(0,*DEFAULT\_HEIGHT\_MID*), **new** Point2D.Double(*DEFAULT\_WIDTH*,*DEFAULT\_HEIGHT\_MID*));

Line2D.Double y\_line = **new** Line2D.Double(**new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, 0), **new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, *DEFAULT\_HEIGHT*));

g2d.draw(x\_line);

g2d.draw(y\_line);

g2d.rotate(Math.*toRadians*(-90),400,400);

g2d.translate(400, 400);

Line2D.Double Line1 = **new** Line2D.Double(**new** Point2D.Double(*x1*, *y1*), **new** Point2D.Double(*x2*,*y2*));

g2d.draw(Line1);

g2d.setColor(Color.***RED***);

Line2D.Double Line2 = **new** Line2D.Double(**new** Point2D.Double(*a1*,*b1*), **new** Point2D.Double(*a2*,*b2*));

g2d.draw(Line2);

}

}

* Output



## About Y=X

* Shahil\_ReflectionXEqualY.java

**import** javax.swing.JFrame;

**public** **class** Shahil\_ReflectionXEqualY {

**static** **int** *arr*[][] = {

{150, 100},

{300, 200}

};

**static** **int**[][] *result* = **new** **int**[2][2];

**final** **static** **int** ***DEFAULT\_HEIGHT*** = 800;

**final** **static** **int** ***DEFAULT\_WIDTH*** = 800;

**final** **static** **int** ***DEFAULT\_X\_MID*** = ***DEFAULT\_HEIGHT***/2;

**final** **static** **int** ***DEFAULT\_Y\_MID*** = ***DEFAULT\_WIDTH***/2;

**public** **static** JFrame *frame*;

**public** Shahil\_ReflectionXEqualY() {

//Frame making

*frame* = **new** JFrame();

*frame*.setTitle("Reflection On Origin");

*frame*.setSize(***DEFAULT\_WIDTH***,***DEFAULT\_HEIGHT***);

*frame*.setVisible(**true**);

*frame*.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

}

**public** **void** Shahil\_drawQuartine(){

//Drawing of axes

*frame*.add(**new** Shahil\_ReflectionXEqualY\_drawComponent(***DEFAULT\_HEIGHT***, ***DEFAULT\_WIDTH***, *arr*, *result*));

}

**public** **void** Shahil\_Opertaion() {

**for**(**int** i = 0 ; i<2; i++) {

*result*[i][0] = *arr*[i][1];

*result*[i][1] = *arr*[i][0];

System.***out***.println(*result*[i][0]+"\t"+*result*[i][1]);

}

}

**public** **static** **void** main(String[] args) {

Shahil\_ReflectionXEqualY Shahil\_obj = **new** Shahil\_ReflectionXEqualY();

Shahil\_obj.Shahil\_Opertaion();

Shahil\_obj.Shahil\_drawQuartine();

}

}

* Shahil\_ReflectionXEqualY\_drawComponent.java

**import** java.awt.Color;

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** java.awt.geom.Line2D;

**import** java.awt.geom.Point2D;

**import** javax.swing.JComponent;

**public** **class** Shahil\_ReflectionXEqualY\_drawComponent **extends** JComponent {

**static** **int** *DEFAULT\_HEIGHT*;

**static** **int** *DEFAULT\_WIDTH*;

**static** **int** *DEFAULT\_HEIGHT\_MID*;

**static** **int** *DEFAULT\_WIDTH\_MID*;

**static** **int** *arr*[][];

**static** **int** *result*[][];

**static** **int** *x1*,*y1*,*x2*,*y2*;

**static** **int** *a1*,*b1*,*a2*,*b2*;

Shahil\_ReflectionXEqualY\_drawComponent(**int** DEFAULT\_HEIGHT, **int** DEFAULT\_WIDTH, **int**[][] arr, **int**[][] result){

**this**.*DEFAULT\_HEIGHT* = DEFAULT\_HEIGHT;

**this**.*DEFAULT\_WIDTH* = DEFAULT\_WIDTH;

*DEFAULT\_HEIGHT\_MID* = DEFAULT\_HEIGHT/2;

*DEFAULT\_WIDTH\_MID* = DEFAULT\_WIDTH/2;

**this**.*arr* = arr;

**this**.*result* = result;

*Shahil\_mapArray*();

}

**public** **static** **void** Shahil\_mapArray() {

*x1* = *arr*[0][0];

*y1* = *arr*[0][1];

*x2* = *arr* [1][0];

*y2* = *arr*[1][1];

*a1* = *result*[0][0];

*b1* = *result*[0][1];

*a2* = *result* [1][0];

*b2* = *result*[1][1];

}

**public** **void** paintComponent(Graphics g){

Graphics2D g2d = (Graphics2D) g;

//Making the x and y axes

Line2D.Double x\_line = **new** Line2D.Double(**new** Point2D.Double(0,*DEFAULT\_HEIGHT\_MID*), **new** Point2D.Double(*DEFAULT\_WIDTH*,*DEFAULT\_HEIGHT\_MID*));

Line2D.Double y\_line = **new** Line2D.Double(**new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, 0), **new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, *DEFAULT\_HEIGHT*));

g2d.draw(x\_line);

g2d.draw(y\_line);

g2d.rotate(Math.*toRadians*(-90),400,400);

g2d.translate(400, 400);

Line2D.Double Line1 = **new** Line2D.Double(**new** Point2D.Double(*x1*, *y1*), **new** Point2D.Double(*x2*,*y2*));

g2d.draw(Line1);

g2d.setColor(Color.***RED***);

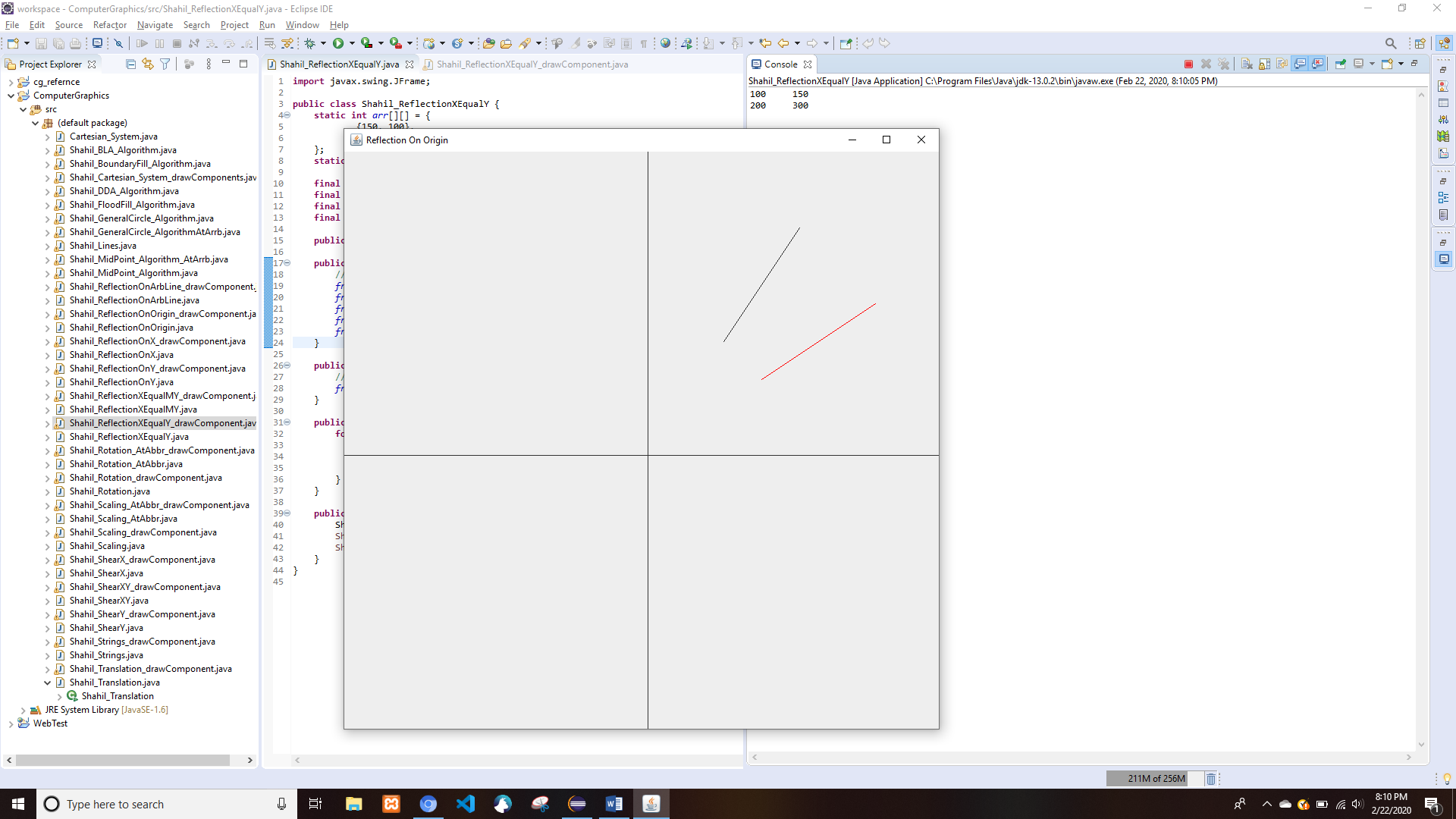
Line2D.Double Line2 = **new** Line2D.Double(**new** Point2D.Double(*a1*,*b1*), **new** Point2D.Double(*a2*,*b2*));

g2d.draw(Line2);

}

}

* Output



## About Y=-X

* Shahil\_ReflectionXEqualMY.java

**import** javax.swing.JFrame;

**public** **class** Shahil\_ReflectionXEqualMY {

**static** **int** *arr*[][] = {

{150, 100},

{300, 200}

};

**static** **int**[][] *result* = **new** **int**[2][2];

**final** **static** **int** ***DEFAULT\_HEIGHT*** = 800;

**final** **static** **int** ***DEFAULT\_WIDTH*** = 800;

**final** **static** **int** ***DEFAULT\_X\_MID*** = ***DEFAULT\_HEIGHT***/2;

**final** **static** **int** ***DEFAULT\_Y\_MID*** = ***DEFAULT\_WIDTH***/2;

**public** **static** JFrame *frame*;

**public** Shahil\_ReflectionXEqualMY() {

//Frame making

*frame* = **new** JFrame();

*frame*.setTitle("Reflection On Origin");

*frame*.setSize(***DEFAULT\_WIDTH***,***DEFAULT\_HEIGHT***);

*frame*.setVisible(**true**);

*frame*.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

}

**public** **void** Shahil\_drawQuartine(){

//Drawing of axes

*frame*.add(**new** Shahil\_ReflectionXEqualY\_drawComponent(***DEFAULT\_HEIGHT***, ***DEFAULT\_WIDTH***, *arr*, *result*));

}

**public** **void** Shahil\_Opertaion() {

**for**(**int** i = 0 ; i<2; i++) {

*result*[i][0] = -*arr*[i][1];

*result*[i][1] = -*arr*[i][0];

System.***out***.println(*result*[i][0]+"\t"+*result*[i][1]);

}

}

**public** **static** **void** main(String[] args) {

Shahil\_ReflectionXEqualMY Shahil\_obj = **new** Shahil\_ReflectionXEqualMY();

Shahil\_obj.Shahil\_Opertaion();

Shahil\_obj.Shahil\_drawQuartine();

}

}

* Shahil\_ReflectionXEqualMY\_drawComponent.java

**import** java.awt.Color;

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** java.awt.geom.Line2D;

**import** java.awt.geom.Point2D;

**import** javax.swing.JComponent;

**public** **class** Shahil\_ReflectionXEqualMY\_drawComponent **extends** JComponent{

**static** **int** *DEFAULT\_HEIGHT*;

**static** **int** *DEFAULT\_WIDTH*;

**static** **int** *DEFAULT\_HEIGHT\_MID*;

**static** **int** *DEFAULT\_WIDTH\_MID*;

**static** **int** *arr*[][];

**static** **int** *result*[][];

**static** **int** *x1*,*y1*,*x2*,*y2*;

**static** **int** *a1*,*b1*,*a2*,*b2*;

Shahil\_ReflectionXEqualMY\_drawComponent(**int** DEFAULT\_HEIGHT, **int** DEFAULT\_WIDTH, **int**[][] arr, **int**[][] result){

**this**.*DEFAULT\_HEIGHT* = DEFAULT\_HEIGHT;

**this**.*DEFAULT\_WIDTH* = DEFAULT\_WIDTH;

*DEFAULT\_HEIGHT\_MID* = DEFAULT\_HEIGHT/2;

*DEFAULT\_WIDTH\_MID* = DEFAULT\_WIDTH/2;

**this**.*arr* = arr;

**this**.*result* = result;

*Shahil\_mapArray*();

}

**public** **static** **void** Shahil\_mapArray() {

*x1* = *arr*[0][0];

*y1* = *arr*[0][1];

*x2* = *arr* [1][0];

*y2* = *arr*[1][1];

*a1* = *result*[0][0];

*b1* = *result*[0][1];

*a2* = *result* [1][0];

*b2* = *result*[1][1];

}

**public** **void** paintComponent(Graphics g){

Graphics2D g2d = (Graphics2D) g;

//Making the x and y axes

Line2D.Double x\_line = **new** Line2D.Double(**new** Point2D.Double(0,*DEFAULT\_HEIGHT\_MID*), **new** Point2D.Double(*DEFAULT\_WIDTH*,*DEFAULT\_HEIGHT\_MID*));

Line2D.Double y\_line = **new** Line2D.Double(**new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, 0), **new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, *DEFAULT\_HEIGHT*));

g2d.draw(x\_line);

g2d.draw(y\_line);

g2d.rotate(Math.*toRadians*(-90),400,400);

g2d.translate(400, 400);

Line2D.Double Line1 = **new** Line2D.Double(**new** Point2D.Double(*x1*, *y1*), **new** Point2D.Double(*x2*,*y2*));

g2d.draw(Line1);

g2d.setColor(Color.***RED***);

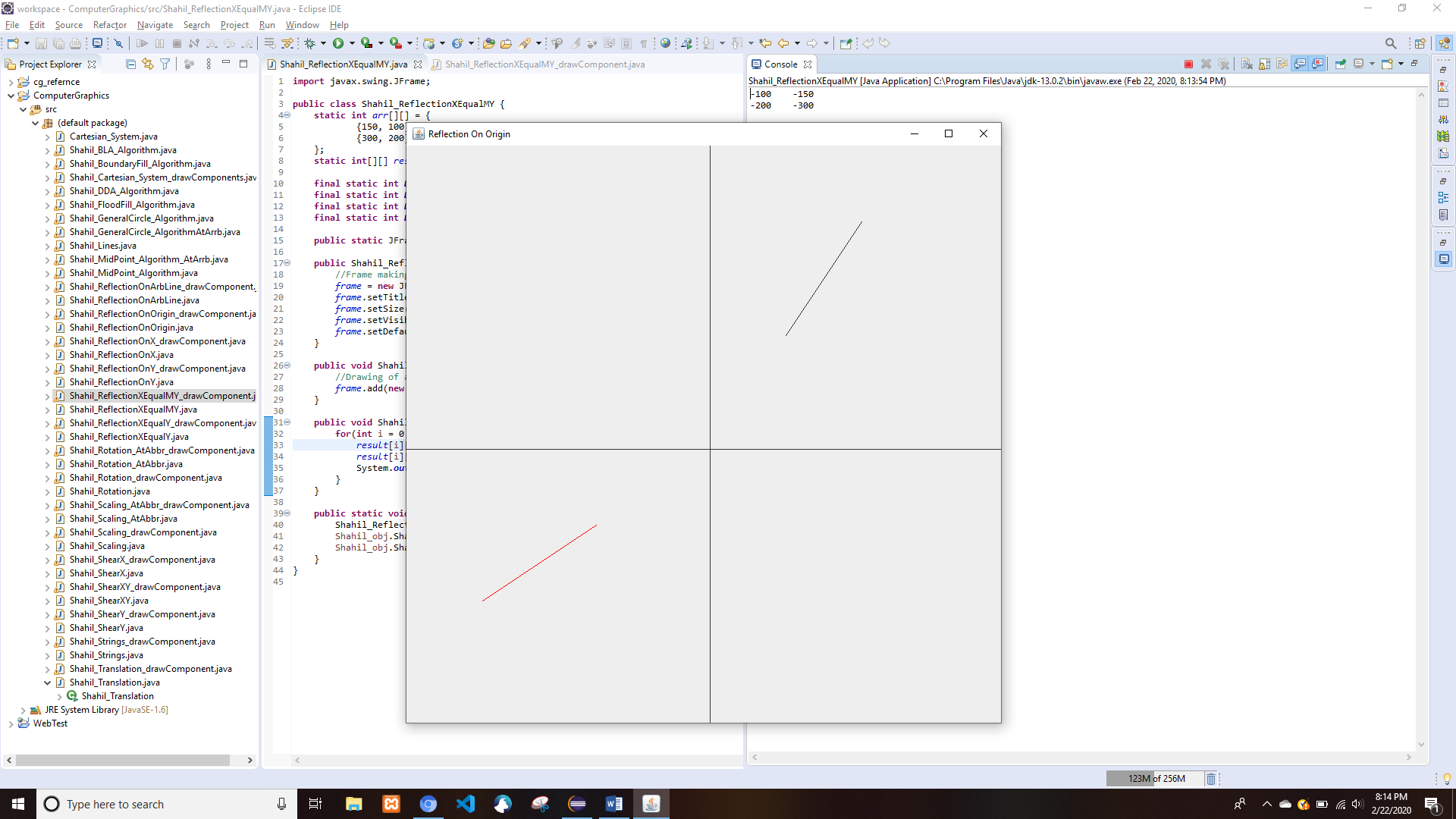
Line2D.Double Line2 = **new** Line2D.Double(**new** Point2D.Double(*a1*,*b1*), **new** Point2D.Double(*a2*,*b2*));

g2d.draw(Line2);

}

}

* Output



## About Origin

* Shahil\_ReflectionOnOrigin.java

**import** javax.swing.JFrame;

**public** **class** Shahil\_ReflectionOnOrigin {

**static** **int** *arr*[][] = {

{100, 100},

{150, 200}

};

**static** **int**[][] *result* = **new** **int**[2][2];

**final** **static** **int** ***DEFAULT\_HEIGHT*** = 800;

**final** **static** **int** ***DEFAULT\_WIDTH*** = 800;

**final** **static** **int** ***DEFAULT\_X\_MID*** = ***DEFAULT\_HEIGHT***/2;

**final** **static** **int** ***DEFAULT\_Y\_MID*** = ***DEFAULT\_WIDTH***/2;

**public** **static** JFrame *frame*;

**public** Shahil\_ReflectionOnOrigin() {

//Frame making

*frame* = **new** JFrame();

*frame*.setTitle("Reflection On Origin");

*frame*.setSize(***DEFAULT\_WIDTH***,***DEFAULT\_HEIGHT***);

*frame*.setVisible(**true**);

*frame*.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

}

**public** **void** Shahil\_drawQuartine(){

//Drawing of axes

*frame*.add(**new** Shahil\_ReflectionOnOrigin\_drawComponent(***DEFAULT\_HEIGHT***, ***DEFAULT\_WIDTH***, *arr*, *result*));

}

**public** **void** Shahil\_Opertaion() {

**for**(**int** i = 0 ; i<2; i++) {

*result*[i][0] = *arr*[i][0] \* -1;

*result*[i][1] = *arr*[i][1] \* -1;

System.***out***.println(*result*[i][0]+"\t"+*result*[i][1]);

}

}

**public** **static** **void** main(String[] args) {

Shahil\_ReflectionOnOrigin Shahil\_obj = **new** Shahil\_ReflectionOnOrigin();

Shahil\_obj.Shahil\_Opertaion();

Shahil\_obj.Shahil\_drawQuartine();

}

}

* Shahil\_ReflectionOnOrigin\_drawComponent.java

**import** java.awt.Color;

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** java.awt.geom.Line2D;

**import** java.awt.geom.Point2D;

**import** javax.swing.JComponent;

**public** **class** Shahil\_ReflectionOnOrigin\_drawComponent **extends** JComponent {

**static** **int** *DEFAULT\_HEIGHT*;

**static** **int** *DEFAULT\_WIDTH*;

**static** **int** *DEFAULT\_HEIGHT\_MID*;

**static** **int** *DEFAULT\_WIDTH\_MID*;

**static** **int** *arr*[][];

**static** **int** *result*[][];

**static** **int** *x1*,*y1*,*x2*,*y2*;

**static** **int** *a1*,*b1*,*a2*,*b2*;

Shahil\_ReflectionOnOrigin\_drawComponent(**int** DEFAULT\_HEIGHT, **int** DEFAULT\_WIDTH, **int**[][] arr, **int**[][] result){

**this**.*DEFAULT\_HEIGHT* = DEFAULT\_HEIGHT;

**this**.*DEFAULT\_WIDTH* = DEFAULT\_WIDTH;

*DEFAULT\_HEIGHT\_MID* = DEFAULT\_HEIGHT/2;

*DEFAULT\_WIDTH\_MID* = DEFAULT\_WIDTH/2;

**this**.*arr* = arr;

**this**.*result* = result;

*Shahil\_mapArray*();

}

**public** **static** **void** Shahil\_mapArray() {

*x1* = *arr*[0][0];

*y1* = *arr*[0][1];

*x2* = *arr* [1][0];

*y2* = *arr*[1][1];

*a1* = *result*[0][0];

*b1* = *result*[0][1];

*a2* = *result* [1][0];

*b2* = *result*[1][1];

}

**public** **void** paintComponent(Graphics g){

Graphics2D g2d = (Graphics2D) g;

//Making the x and y axes

Line2D.Double x\_line = **new** Line2D.Double(**new** Point2D.Double(0,*DEFAULT\_HEIGHT\_MID*), **new** Point2D.Double(*DEFAULT\_WIDTH*,*DEFAULT\_HEIGHT\_MID*));

Line2D.Double y\_line = **new** Line2D.Double(**new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, 0), **new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, *DEFAULT\_HEIGHT*));

g2d.draw(x\_line);

g2d.draw(y\_line);

g2d.rotate(Math.*toRadians*(-90),400,400);

g2d.translate(400, 400);

Line2D.Double Line1 = **new** Line2D.Double(**new** Point2D.Double(*x1*, *y1*), **new** Point2D.Double(*x2*,*y2*));

g2d.draw(Line1);

g2d.setColor(Color.***RED***);

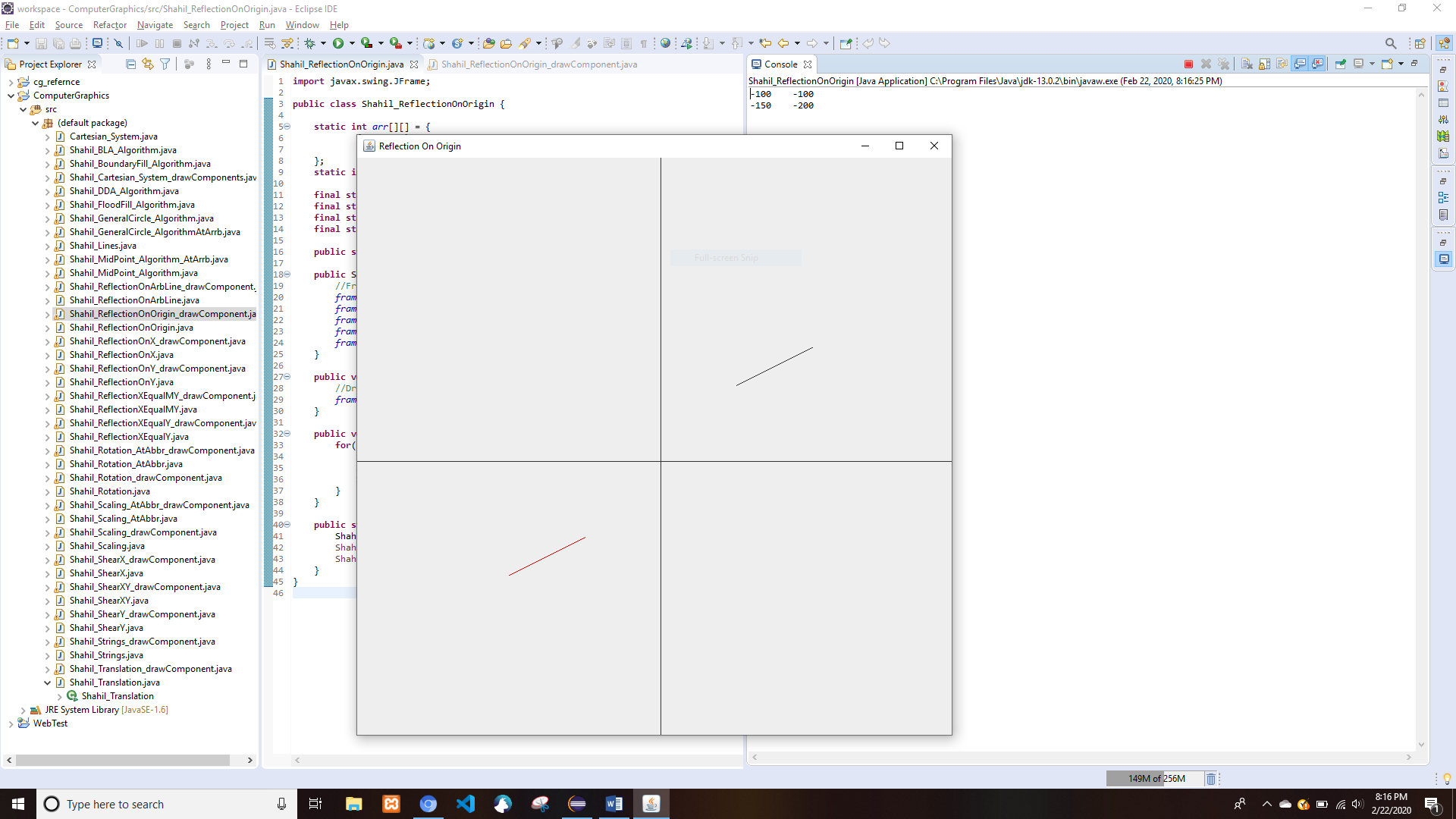
Line2D.Double Line2 = **new** Line2D.Double(**new** Point2D.Double(*a1*,*b1*), **new** Point2D.Double(*a2*,*b2*));

g2d.draw(Line2);

}

}

* Output



## About an Arbitrary Line

* Shahil\_ReflectionOnArbLine.java

**import** javax.swing.JFrame;

//reflecting on arbitrary line with the equation y = m + 50

**public** **class** Shahil\_ReflectionOnArbLine {

**static** **int** *arr*[][] = {

{90, 100},

{200, 300}

};

**static** **int**[][] *result* = **new** **int**[2][2];

**static** **int** *c* = 100;

**static** **int**[] *translate\_factor* = {0, *c*};

**final** **static** **int** ***DEFAULT\_HEIGHT*** = 800;

**final** **static** **int** ***DEFAULT\_WIDTH*** = 800;

**final** **static** **int** ***DEFAULT\_X\_MID*** = ***DEFAULT\_HEIGHT***/2;

**final** **static** **int** ***DEFAULT\_Y\_MID*** = ***DEFAULT\_WIDTH***/2;

**public** **static** JFrame *frame*;

**public** Shahil\_ReflectionOnArbLine() {

//Frame making

*frame* = **new** JFrame();

*frame*.setTitle("Reflection On Origin");

*frame*.setSize(***DEFAULT\_WIDTH***,***DEFAULT\_HEIGHT***);

*frame*.setVisible(**true**);

*frame*.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

}

**public** **void** Shahil\_drawQuartine(){

//Drawing of axes

*frame*.add(**new** Shahil\_ReflectionOnArbLine\_drawComponent(***DEFAULT\_HEIGHT***, ***DEFAULT\_WIDTH***, *arr*, *result*));

}

**public** **void** Shahil\_Translate(**int** a, **int** b) {

**for**(**int** i = 0 ; i<2; i++) {

*arr*[i][0] = *arr*[i][0] + a;

*arr*[i][1] = *arr*[i][1] + b;

}

}

**public** **void** Shahil\_Rotate(**double** m) {

**double** angleRadian= Math.*toRadians*(m);

**for**(**int** i = 0 ; i<2; i++) {

*arr*[i][0]=(**int**)Math.*round*(*arr*[i][0]\*Math.*cos*(angleRadian)+*arr*[i][1]\*Math.*sin*(angleRadian));

*arr*[i][1]=(**int**)Math.*round*(*arr*[i][0]\*Math.*sin*(angleRadian)+*arr*[i][1]\*Math.*cos*(angleRadian));

}

}

**public** **void** Shahil\_Opertaion() {

**double** m = Math.*toRadians*(Math.*atan2*((*arr*[1][1]-*arr*[0][1]),(*arr*[1][0]-*arr*[0][0])));

**double** ang = Math.*toRadians*(Math.*atan*(m));

Shahil\_Translate(0, -*c*);

Shahil\_Rotate(ang);

**for**(**int** i = 0 ; i<2; i++) {

*result*[i][0] = *arr*[i][0] \* 1;

*result*[i][1] = *arr*[i][1] \* -1;

// System.out.println(result[i][0]+"\t"+result[i][1]);

}

Shahil\_Rotate(-ang);

Shahil\_Translate(0, *c*);

}

**public** **static** **void** main(String[] args) {

Shahil\_ReflectionOnArbLine Shahil\_obj = **new** Shahil\_ReflectionOnArbLine();

Shahil\_obj.Shahil\_Opertaion();

Shahil\_obj.Shahil\_drawQuartine();

}

}

* Shahil\_ReflectionOnArbLine\_drawComponent.java

**import** java.awt.Color;

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** java.awt.geom.Line2D;

**import** java.awt.geom.Point2D;

**import** javax.swing.JComponent;

**public** **class** Shahil\_ReflectionOnArbLine\_drawComponent **extends** JComponent{

**static** **int** *DEFAULT\_HEIGHT*;

**static** **int** *DEFAULT\_WIDTH*;

**static** **int** *DEFAULT\_HEIGHT\_MID*;

**static** **int** *DEFAULT\_WIDTH\_MID*;

**static** **int** *arr*[][];

**static** **int** *result*[][];

**static** **int** *x1*,*y1*,*x2*,*y2*;

**static** **int** *a1*,*b1*,*a2*,*b2*;

Shahil\_ReflectionOnArbLine\_drawComponent(**int** DEFAULT\_HEIGHT, **int** DEFAULT\_WIDTH, **int**[][] arr, **int**[][] result){

**this**.*DEFAULT\_HEIGHT* = DEFAULT\_HEIGHT;

**this**.*DEFAULT\_WIDTH* = DEFAULT\_WIDTH;

*DEFAULT\_HEIGHT\_MID* = DEFAULT\_HEIGHT/2;

*DEFAULT\_WIDTH\_MID* = DEFAULT\_WIDTH/2;

**this**.*arr* = arr;

**this**.*result* = result;

*Shahil\_mapArray*();

}

**public** **static** **void** Shahil\_mapArray() {

*x1* = *arr*[0][0];

*y1* = *arr*[0][1];

*x2* = *arr* [1][0];

*y2* = *arr*[1][1];

*a1* = *result*[0][0];

*b1* = *result*[0][1];

*a2* = *result* [1][0];

*b2* = *result*[1][1];

}

**public** **void** paintComponent(Graphics g){

Graphics2D g2d = (Graphics2D) g;

//Making the x and y axes

Line2D.Double x\_line = **new** Line2D.Double(**new** Point2D.Double(0,*DEFAULT\_HEIGHT\_MID*), **new** Point2D.Double(*DEFAULT\_WIDTH*,*DEFAULT\_HEIGHT\_MID*));

Line2D.Double y\_line = **new** Line2D.Double(**new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, 0), **new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, *DEFAULT\_HEIGHT*));

g2d.draw(x\_line);

g2d.draw(y\_line);

g2d.rotate(Math.*toRadians*(-90),400,400);

g2d.translate(400, 400);

Line2D.Double Line1 = **new** Line2D.Double(**new** Point2D.Double(*x1*, *y1*), **new** Point2D.Double(*x2*,*y2*));

g2d.draw(Line1);

g2d.setColor(Color.***RED***);

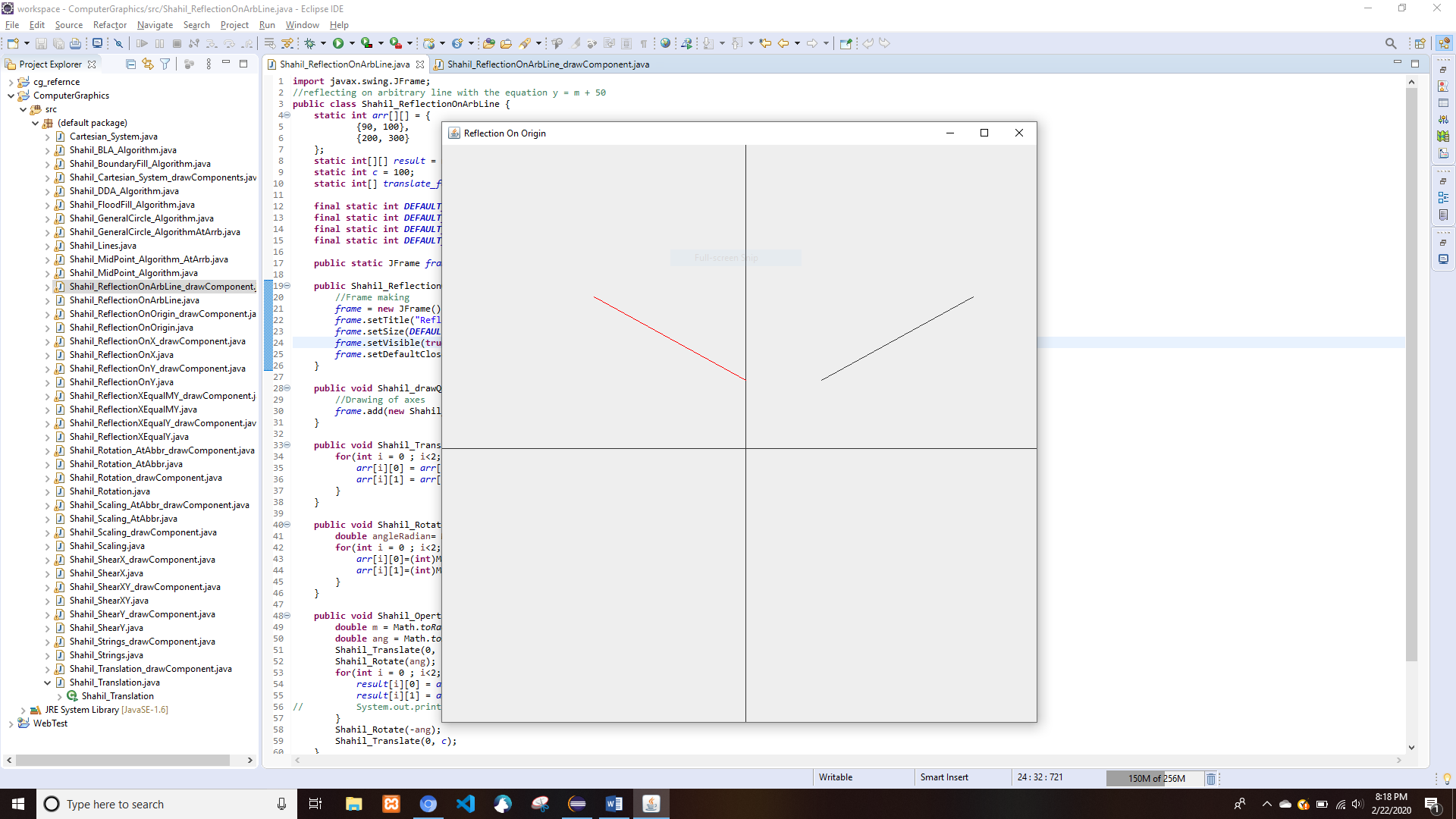
Line2D.Double Line2 = **new** Line2D.Double(**new** Point2D.Double(*a1*,*b1*), **new** Point2D.Double(*a2*,*b2*));

g2d.draw(Line2);

}

}

* Output



# Shear

## In X Direction

* Shahil\_ShearX.java

**import** javax.swing.JFrame;

**public** **class** Shahil\_ShearX {

**static** **int** *arr\_x*[] = {10,10,200};

**static** **int** *arr\_y*[] = {80,150,0};

**static** **int**[] *res\_x* = **new** **int**[3];

**static** **int**[] *res\_y* = **new** **int**[3];

**int** shy = 2;

**final** **static** **int** ***DEFAULT\_HEIGHT*** = 800;

**final** **static** **int** ***DEFAULT\_WIDTH*** = 800;

**final** **static** **int** ***DEFAULT\_X\_MID*** = ***DEFAULT\_HEIGHT***/2;

**final** **static** **int** ***DEFAULT\_Y\_MID*** = ***DEFAULT\_WIDTH***/2;

**public** **static** JFrame *frame*;

**public** Shahil\_ShearX() {

//Frame making

*frame* = **new** JFrame();

*frame*.setTitle("X-Direction Shear of a triangle");

*frame*.setSize(***DEFAULT\_WIDTH***,***DEFAULT\_HEIGHT***);

*frame*.setVisible(**true**);

*frame*.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

}

**public** **void** Shahil\_drawQuartine(){

//Drawing of axes

*frame*.add(**new** Shahil\_ShearX\_drawComponent(***DEFAULT\_HEIGHT***, ***DEFAULT\_WIDTH***, *arr\_x*, *arr\_y* , *res\_x*, *res\_y*));

}

**public** **void** Shahil\_Opertaion() {

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

*res\_x*[i] = *arr\_x*[i];

*res\_y*[i] = *arr\_y*[i] + shy \* *arr\_x*[i];

System.***out***.println(*res\_x*[i]+"\t"+*res\_y*[i]);

}

}

**public** **static** **void** main(String[] args) {

Shahil\_ShearX Shahil\_obj = **new** Shahil\_ShearX();

Shahil\_obj.Shahil\_Opertaion();

Shahil\_obj.Shahil\_drawQuartine();

}

}

* Shahil\_ShearX\_drawComponent.java

**import** java.awt.Color;

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** java.awt.geom.Line2D;

**import** java.awt.geom.Point2D;

**import** javax.swing.JComponent;

**public** **class** Shahil\_ShearX\_drawComponent **extends** JComponent{

**static** **int** *DEFAULT\_HEIGHT*;

**static** **int** *DEFAULT\_WIDTH*;

**static** **int** *DEFAULT\_HEIGHT\_MID*;

**static** **int** *DEFAULT\_WIDTH\_MID*;

**static** **int** *arr\_x*[];

**static** **int** *arr\_y*[];

**static** **int** *res\_x*[];

**static** **int** *res\_y*[];

**static** **int** *x1*, *x2*, *x3*;

**static** **int** *y1*, *y2*, *y3*;

Shahil\_ShearX\_drawComponent(**int** DEFAULT\_HEIGHT, **int** DEFAULT\_WIDTH, **int**[] arr\_x, **int**[] arr\_y, **int**[] res\_x, **int**[] res\_y){

**this**.*DEFAULT\_HEIGHT* = DEFAULT\_HEIGHT;

**this**.*DEFAULT\_WIDTH* = DEFAULT\_WIDTH;

*DEFAULT\_HEIGHT\_MID* = DEFAULT\_HEIGHT/2;

*DEFAULT\_WIDTH\_MID* = DEFAULT\_WIDTH/2;

**this**.*arr\_x* = arr\_x;

**this**.*arr\_y* = arr\_y;

**this**.*res\_x* = res\_x;

**this**.*res\_y* = res\_y;

}

**public** **void** paintComponent(Graphics g){

Graphics2D g2d = (Graphics2D) g;

//Making the x and y axes

Line2D.Double x\_line = **new** Line2D.Double(**new** Point2D.Double(0,*DEFAULT\_HEIGHT\_MID*), **new** Point2D.Double(*DEFAULT\_WIDTH*,*DEFAULT\_HEIGHT\_MID*));

Line2D.Double y\_line = **new** Line2D.Double(**new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, 0), **new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, *DEFAULT\_HEIGHT*));

g2d.draw(x\_line);

g2d.draw(y\_line);

g2d.rotate(Math.*toRadians*(-90),400,400);

g2d.translate(400, 400);

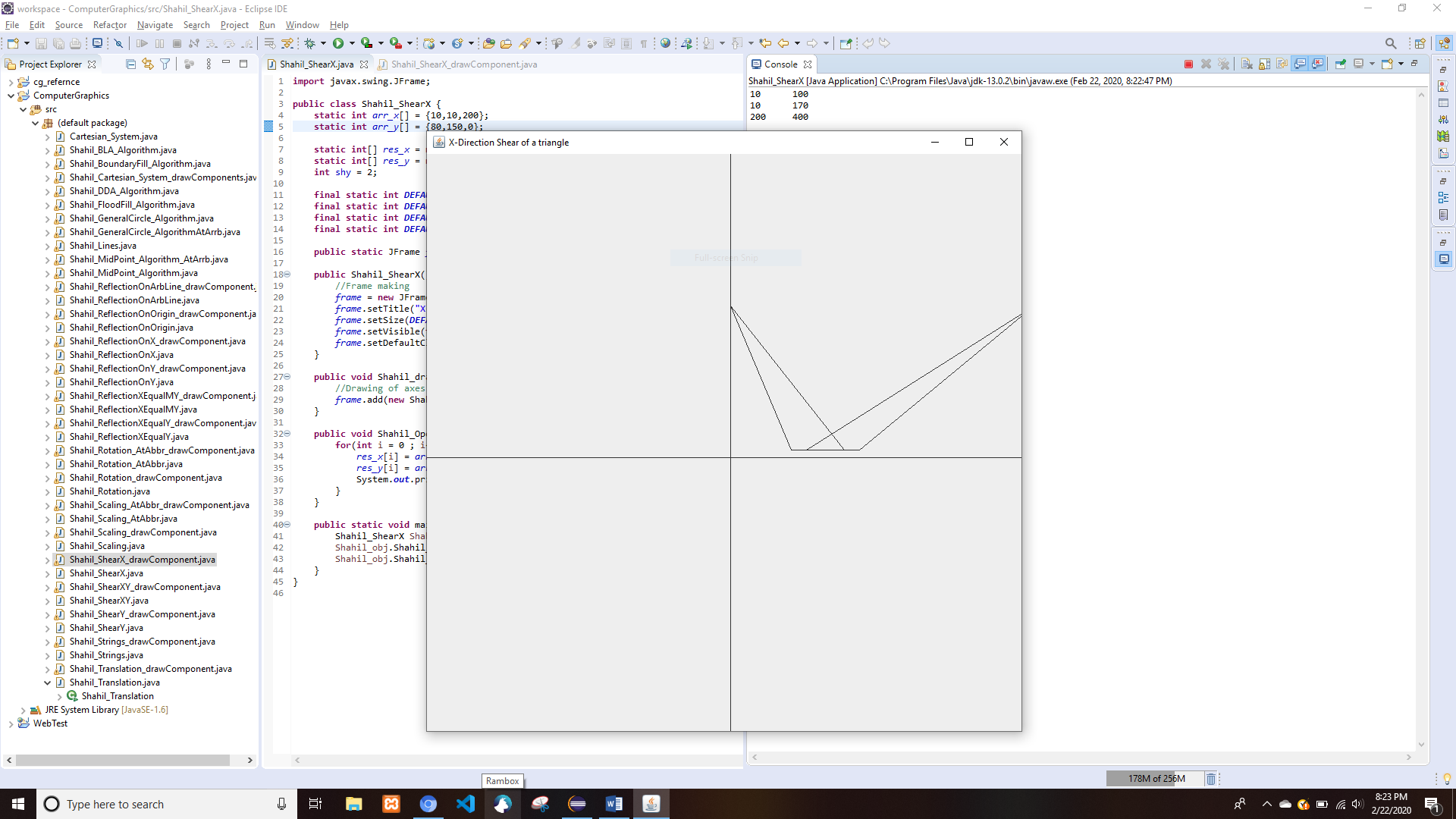
g2d.drawPolygon(*arr\_x*, *arr\_y*, *arr\_x*.length);

g2d.drawPolygon(*res\_x*, *res\_y*, *res\_x*.length);

}

}

* Output



## In Y Direction

* Shahil\_ShearY.java

**import** javax.swing.JFrame;

**public** **class** Shahil\_ShearY {

**static** **int** *arr\_x*[] = {10,10,200};

**static** **int** *arr\_y*[] = {80,150,0};

**static** **int**[] *res\_x* = **new** **int**[3];

**static** **int**[] *res\_y* = **new** **int**[3];

**int** shx = 2;

**final** **static** **int** ***DEFAULT\_HEIGHT*** = 800;

**final** **static** **int** ***DEFAULT\_WIDTH*** = 800;

**final** **static** **int** ***DEFAULT\_X\_MID*** = ***DEFAULT\_HEIGHT***/2;

**final** **static** **int** ***DEFAULT\_Y\_MID*** = ***DEFAULT\_WIDTH***/2;

**public** **static** JFrame *frame*;

**public** Shahil\_ShearY() {

//Frame making

*frame* = **new** JFrame();

*frame*.setTitle("Y-Direction Shear of a triangle");

*frame*.setSize(***DEFAULT\_WIDTH***,***DEFAULT\_HEIGHT***);

*frame*.setVisible(**true**);

*frame*.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

}

**public** **void** Shahil\_drawQuartine(){

//Drawing of axes

*frame*.add(**new** Shahil\_ShearY\_drawComponent(***DEFAULT\_HEIGHT***, ***DEFAULT\_WIDTH***, *arr\_x*, *arr\_y* , *res\_x*, *res\_y*));

}

**public** **void** Shahil\_Opertaion() {

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

*res\_x*[i] = *arr\_x*[i] + shx \* *arr\_y*[i];

*res\_y*[i] = *arr\_y*[i];

System.***out***.println(*res\_x*[i]+"\t"+*res\_y*[i]);

}

}

**public** **static** **void** main(String[] args) {

Shahil\_ShearY Shahil\_obj = **new** Shahil\_ShearY();

Shahil\_obj.Shahil\_Opertaion();

Shahil\_obj.Shahil\_drawQuartine();

}

}

* Shahil\_ShearY\_drawComponent.java

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** java.awt.geom.Line2D;

**import** java.awt.geom.Point2D;

**import** javax.swing.JComponent;

**public** **class** Shahil\_ShearY\_drawComponent **extends** JComponent{

**static** **int** *DEFAULT\_HEIGHT*;

**static** **int** *DEFAULT\_WIDTH*;

**static** **int** *DEFAULT\_HEIGHT\_MID*;

**static** **int** *DEFAULT\_WIDTH\_MID*;

**static** **int** *arr\_x*[];

**static** **int** *arr\_y*[];

**static** **int** *res\_x*[];

**static** **int** *res\_y*[];

**static** **int** *x1*, *x2*, *x3*;

**static** **int** *y1*, *y2*, *y3*;

Shahil\_ShearY\_drawComponent(**int** DEFAULT\_HEIGHT, **int** DEFAULT\_WIDTH, **int**[] arr\_x, **int**[] arr\_y, **int**[] res\_x, **int**[] res\_y){

**this**.*DEFAULT\_HEIGHT* = DEFAULT\_HEIGHT;

**this**.*DEFAULT\_WIDTH* = DEFAULT\_WIDTH;

*DEFAULT\_HEIGHT\_MID* = DEFAULT\_HEIGHT/2;

*DEFAULT\_WIDTH\_MID* = DEFAULT\_WIDTH/2;

**this**.*arr\_x* = arr\_x;

**this**.*arr\_y* = arr\_y;

**this**.*res\_x* = res\_x;

**this**.*res\_y* = res\_y;

}

**public** **void** paintComponent(Graphics g){

Graphics2D g2d = (Graphics2D) g;

//Making the x and y axes

Line2D.Double x\_line = **new** Line2D.Double(**new** Point2D.Double(0,*DEFAULT\_HEIGHT\_MID*), **new** Point2D.Double(*DEFAULT\_WIDTH*,*DEFAULT\_HEIGHT\_MID*));

Line2D.Double y\_line = **new** Line2D.Double(**new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, 0), **new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, *DEFAULT\_HEIGHT*));

g2d.draw(x\_line);

g2d.draw(y\_line);

g2d.rotate(Math.*toRadians*(-90),400,400);

g2d.translate(400, 400);

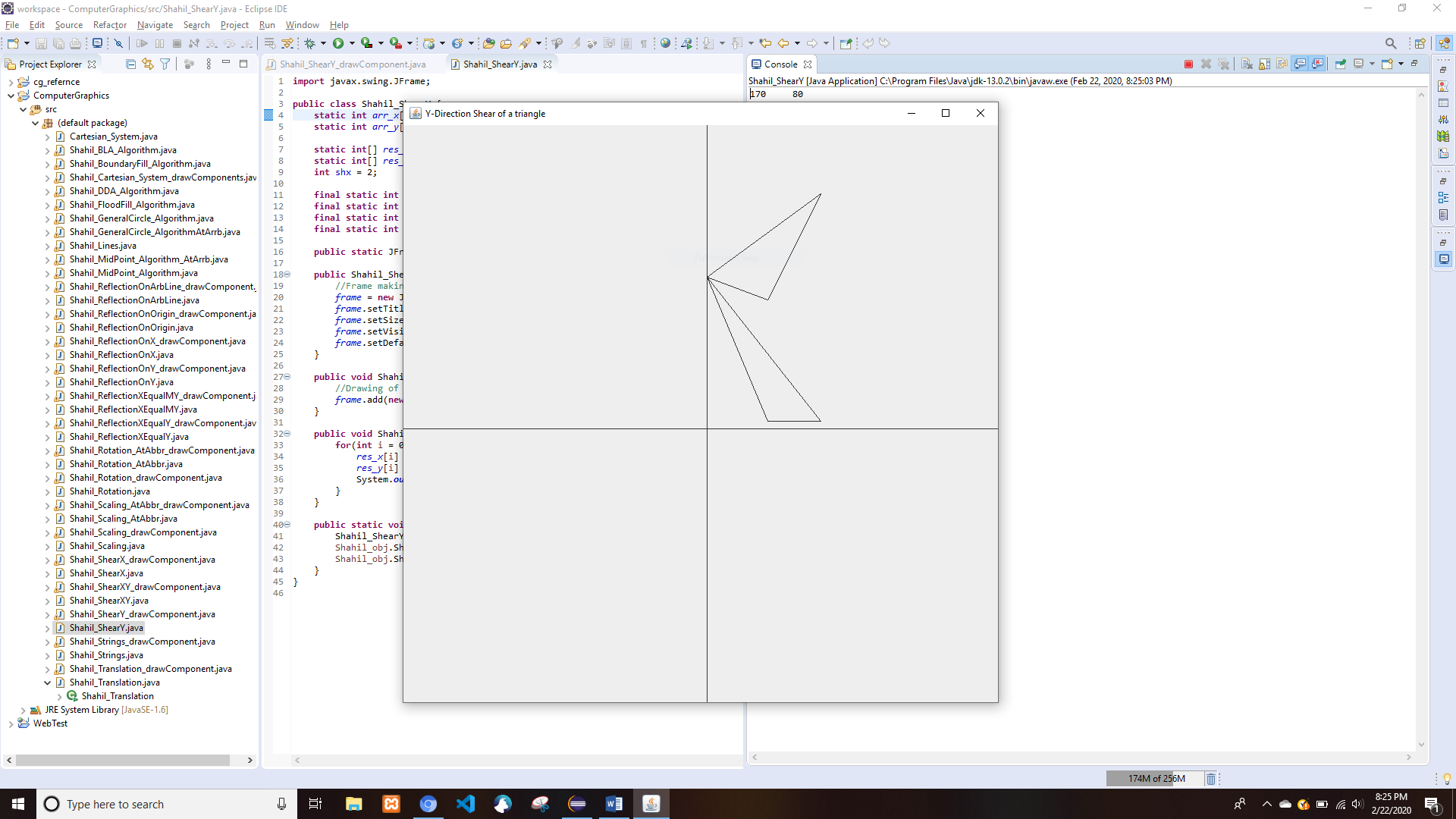
g2d.drawPolygon(*arr\_x*, *arr\_y*, *arr\_x*.length);

g2d.drawPolygon(*res\_x*, *res\_y*, *res\_x*.length);

}

}

* Output



## In XY Direction

* Shahil\_ShearXY.java

**import** javax.swing.JFrame;

**public** **class** Shahil\_ShearXY {

**static** **int** *arr\_x*[] = {10,10,150};

**static** **int** *arr\_y*[] = {80,150,0};

**static** **int**[] *res\_x* = **new** **int**[3];

**static** **int**[] *res\_y* = **new** **int**[3];

**int** shx = 2;

**int** shy = 2;

**final** **static** **int** ***DEFAULT\_HEIGHT*** = 800;

**final** **static** **int** ***DEFAULT\_WIDTH*** = 800;

**final** **static** **int** ***DEFAULT\_X\_MID*** = ***DEFAULT\_HEIGHT***/2;

**final** **static** **int** ***DEFAULT\_Y\_MID*** = ***DEFAULT\_WIDTH***/2;

**public** **static** JFrame *frame*;

**public** Shahil\_ShearXY() {

//Frame making

*frame* = **new** JFrame();

*frame*.setTitle("Y-Direction Shear of a triangle");

*frame*.setSize(***DEFAULT\_WIDTH***,***DEFAULT\_HEIGHT***);

*frame*.setVisible(**true**);

*frame*.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

}

**public** **void** Shahil\_drawQuartine(){

//Drawing of axes

*frame*.add(**new** Shahil\_ShearXY\_drawComponent(***DEFAULT\_HEIGHT***, ***DEFAULT\_WIDTH***, *arr\_x*, *arr\_y* , *res\_x*, *res\_y*));

}

**public** **void** Shahil\_Opertaion() {

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

*res\_x*[i] = *arr\_x*[i] + shx \* *arr\_y*[i];

*res\_y*[i] = *arr\_y*[i] + shy \* *arr\_x*[i];

System.***out***.println(*res\_x*[i]+"\t"+*res\_y*[i]);

}

}

**public** **static** **void** main(String[] args) {

Shahil\_ShearXY Shahil\_obj = **new** Shahil\_ShearXY();

Shahil\_obj.Shahil\_Opertaion();

Shahil\_obj.Shahil\_drawQuartine();

}

}

* Shahil\_ShearXY\_drawComponent.java

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** java.awt.geom.Line2D;

**import** java.awt.geom.Point2D;

**import** javax.swing.JComponent;

**public** **class** Shahil\_ShearXY\_drawComponent **extends** JComponent{

**static** **int** *DEFAULT\_HEIGHT*;

**static** **int** *DEFAULT\_WIDTH*;

**static** **int** *DEFAULT\_HEIGHT\_MID*;

**static** **int** *DEFAULT\_WIDTH\_MID*;

**static** **int** *arr\_x*[];

**static** **int** *arr\_y*[];

**static** **int** *res\_x*[];

**static** **int** *res\_y*[];

**static** **int** *x1*, *x2*, *x3*;

**static** **int** *y1*, *y2*, *y3*;

Shahil\_ShearXY\_drawComponent(**int** DEFAULT\_HEIGHT, **int** DEFAULT\_WIDTH, **int**[] arr\_x, **int**[] arr\_y, **int**[] res\_x, **int**[] res\_y){

**this**.*DEFAULT\_HEIGHT* = DEFAULT\_HEIGHT;

**this**.*DEFAULT\_WIDTH* = DEFAULT\_WIDTH;

*DEFAULT\_HEIGHT\_MID* = DEFAULT\_HEIGHT/2;

*DEFAULT\_WIDTH\_MID* = DEFAULT\_WIDTH/2;

**this**.*arr\_x* = arr\_x;

**this**.*arr\_y* = arr\_y;

**this**.*res\_x* = res\_x;

**this**.*res\_y* = res\_y;

}

**public** **void** paintComponent(Graphics g){

Graphics2D g2d = (Graphics2D) g;

//Making the x and y axes

Line2D.Double x\_line = **new** Line2D.Double(**new** Point2D.Double(0,*DEFAULT\_HEIGHT\_MID*), **new** Point2D.Double(*DEFAULT\_WIDTH*,*DEFAULT\_HEIGHT\_MID*));

Line2D.Double y\_line = **new** Line2D.Double(**new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, 0), **new** Point2D.Double(*DEFAULT\_WIDTH\_MID*, *DEFAULT\_HEIGHT*));

g2d.draw(x\_line);

g2d.draw(y\_line);

g2d.rotate(Math.*toRadians*(-90),400,400);

g2d.translate(400, 400);

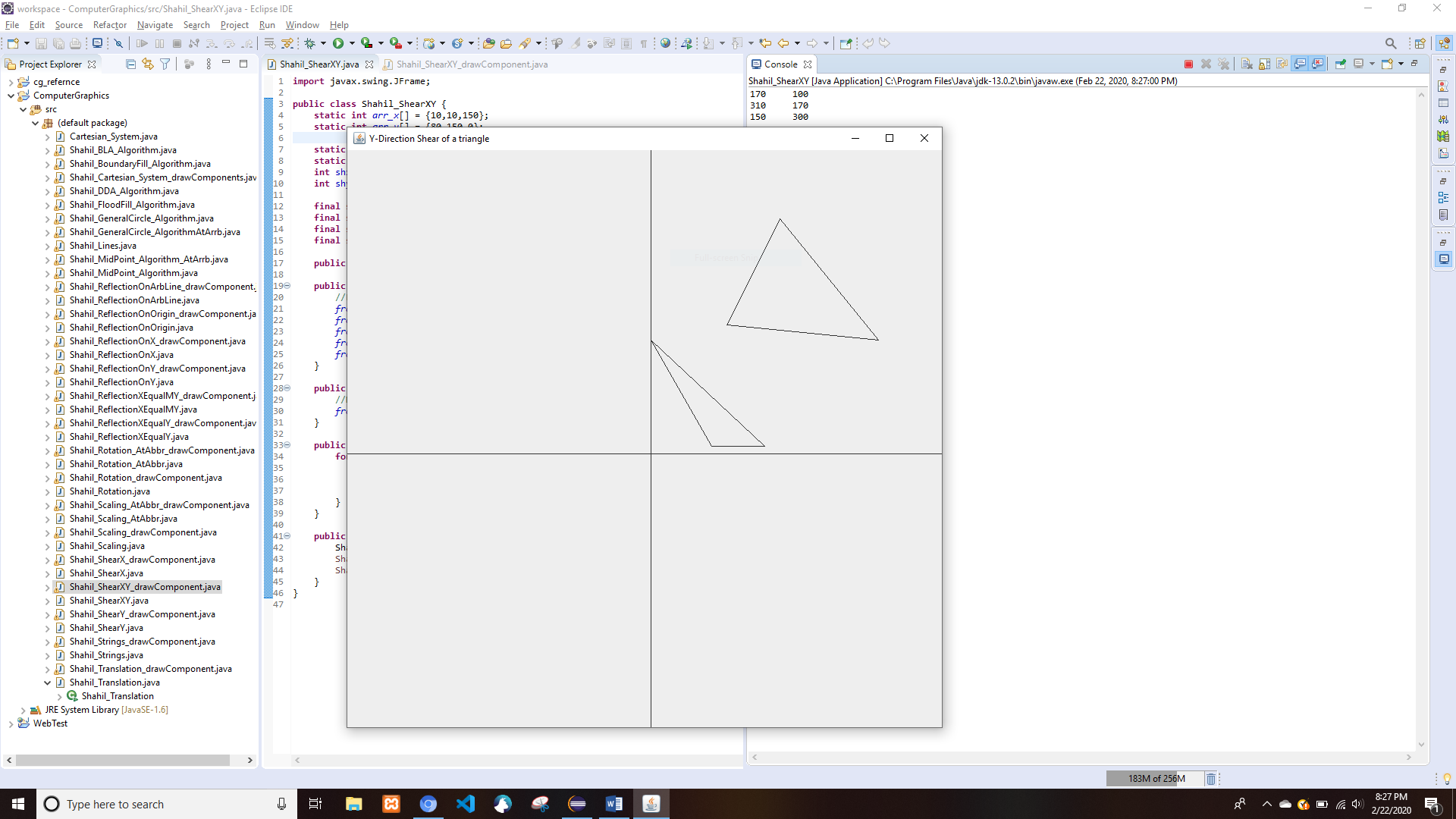
g2d.drawPolygon(*arr\_x*, *arr\_y*, *arr\_x*.length);

g2d.drawPolygon(*res\_x*, *res\_y*, *res\_x*.length);

}

}

* Output



# Drawing Lines

* Shahil\_Lines.java

**import** java.awt.BasicStroke;

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** javax.swing.JComponent;

**import** javax.swing.JFrame;

**class** Shahil\_MyCanvas **extends** JComponent {

**public** **void** paint(Graphics g) {

Graphics2D g2d = (Graphics2D) g;

**float**[] dash1 = { 2f, 0f, 2f };

g2d.drawLine(200, 300, 600, 300);

BasicStroke bs1 = **new** BasicStroke(1,

BasicStroke.***CAP\_BUTT***,

BasicStroke.***JOIN\_ROUND***,

1.0f,

dash1,

2f);

g2d.setStroke(bs1);

g2d.drawLine(200, 200, 600, 200);

BasicStroke bs2=**new** BasicStroke(30);

g2d.setStroke(bs2);

g2d.drawLine(200, 500, 600, 500);

}

}

**public** **class** Shahil\_Lines {

**public** **static** **void** main(String[] args) {

JFrame frame = **new** JFrame();

frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

frame.setSize(800,800);

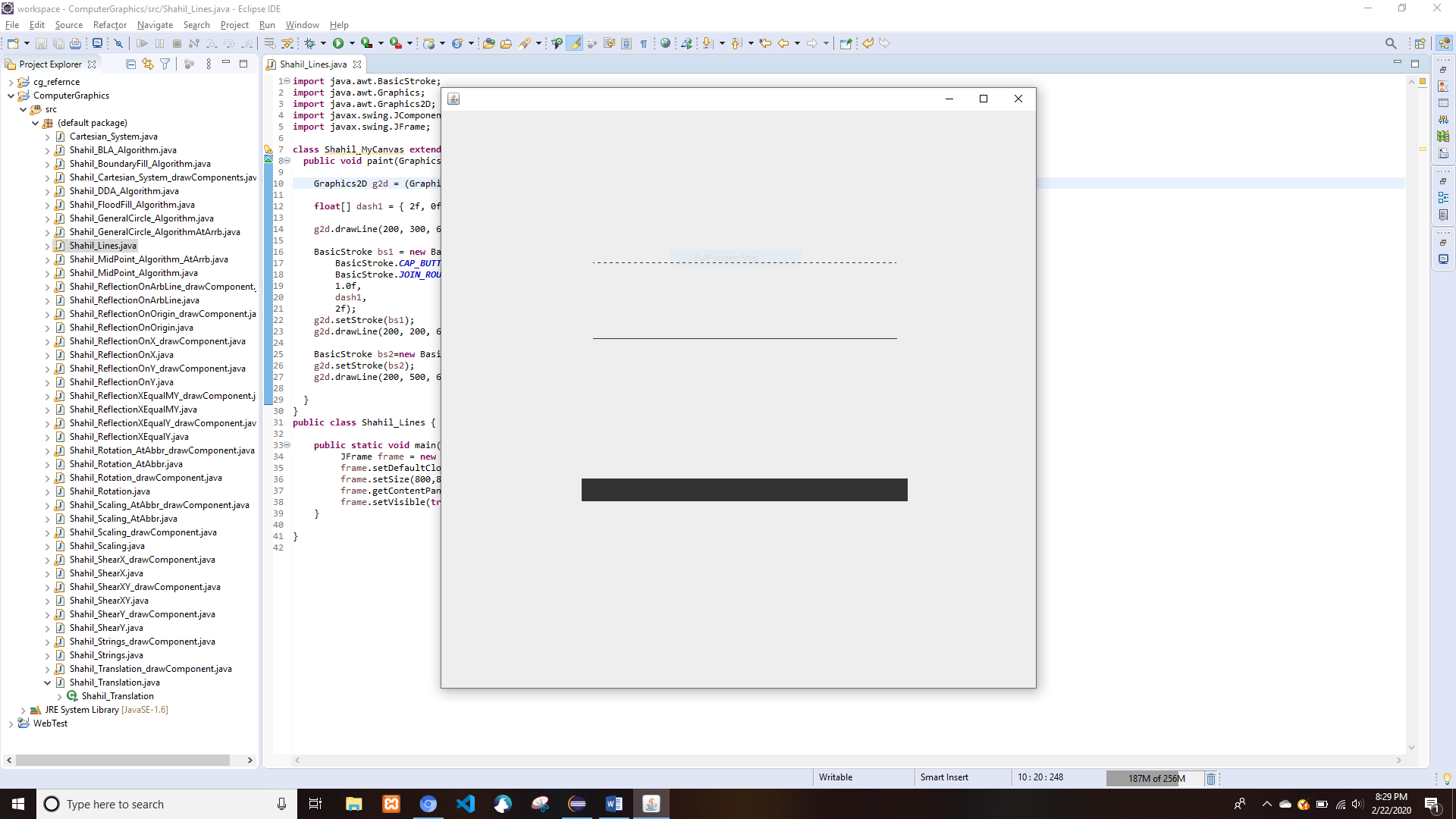
frame.getContentPane().add(**new** Shahil\_MyCanvas());

frame.setVisible(**true**);

}

}

* Output



# Drawing Strings

* Shahil\_Strings.java

**import** javax.swing.JFrame;

**public** **class** Shahil\_Strings {

JFrame frame;

String str;

Shahil\_Strings(){

//Frame making

frame = **new** JFrame();

frame.setTitle("Reflection On Origin");

frame.setSize(800,800);

frame.setVisible(**true**);

frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

}

**public** **void** Shahil\_draw(){

//Drawing of axes

frame.add(**new** Shahil\_Strings\_drawComponent(str));

}

**public** **void** Shahil\_Opertaion() {

str = "Report of Computer Graphics";

}

**public** **static** **void** main(String[] args) {

Shahil\_Strings Shahil\_obj = **new** Shahil\_Strings();

Shahil\_obj.Shahil\_Opertaion();

Shahil\_obj.Shahil\_draw();

}

}

* Output



# Bresenham's Line Algorithm(BLA)

* Shahil\_BLA\_Algorithm.java

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** javax.swing.JComponent;

**import** javax.swing.JFrame;

**class** Shahil\_BLACanvas **extends** JComponent {

**int** x1,x2,y1,y2;

**int** a,b;

**int** dx,dy;//double Steps;

**int** x,y;//double xINC,yINC;

**int** p;

**public** **void** paint(Graphics g) {

Graphics2D g2d = (Graphics2D) g;

x1=100;

y1=400;

x2=700;

y2=400;

**if**(x2>x1)

a=1;

**else**

a=-1;

**if**(y2>y1)

b=1;

**else**

b=-1;

dx=Math.*abs*(x2-x1);

dy=Math.*abs*(y2-y1);

x=x1;

y=y1;

**if**(dx>dy){

p=((2\*dy-dx));

g.fillOval(x,y,2,2);

**for**(**int** i=0;i<dx;i++){

**if**(p<=0){

x=x+a;

y=y;

p= (p+2\*dy);

}**else**{

x=x+a;

y=y+b;

p=(p+2\*dy-2\*dx);

}

g.fillOval(x,y,2,2);

}

}

**else**{

p=((2\*dx-dy));

g.fillOval(x, y, 2, 2);

**for**(**int** i=1;i<=dy;i++){

**if**(p<=0){

x=x;

y=y+b;

p= (p+2\*dx);

}**else**{

x=x+a;

y=y+b;

p= (p+2\*dx-2\*dy);

}

g.fillOval(x, y, 2, 2);

}

}

}

}

**public** **class** Shahil\_BLA\_Algorithm {

**public** **static** **void** main(String[] args) {

JFrame frame = **new** JFrame();

frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

frame.setSize(800,800);

frame.getContentPane().add(**new** Shahil\_BLACanvas());

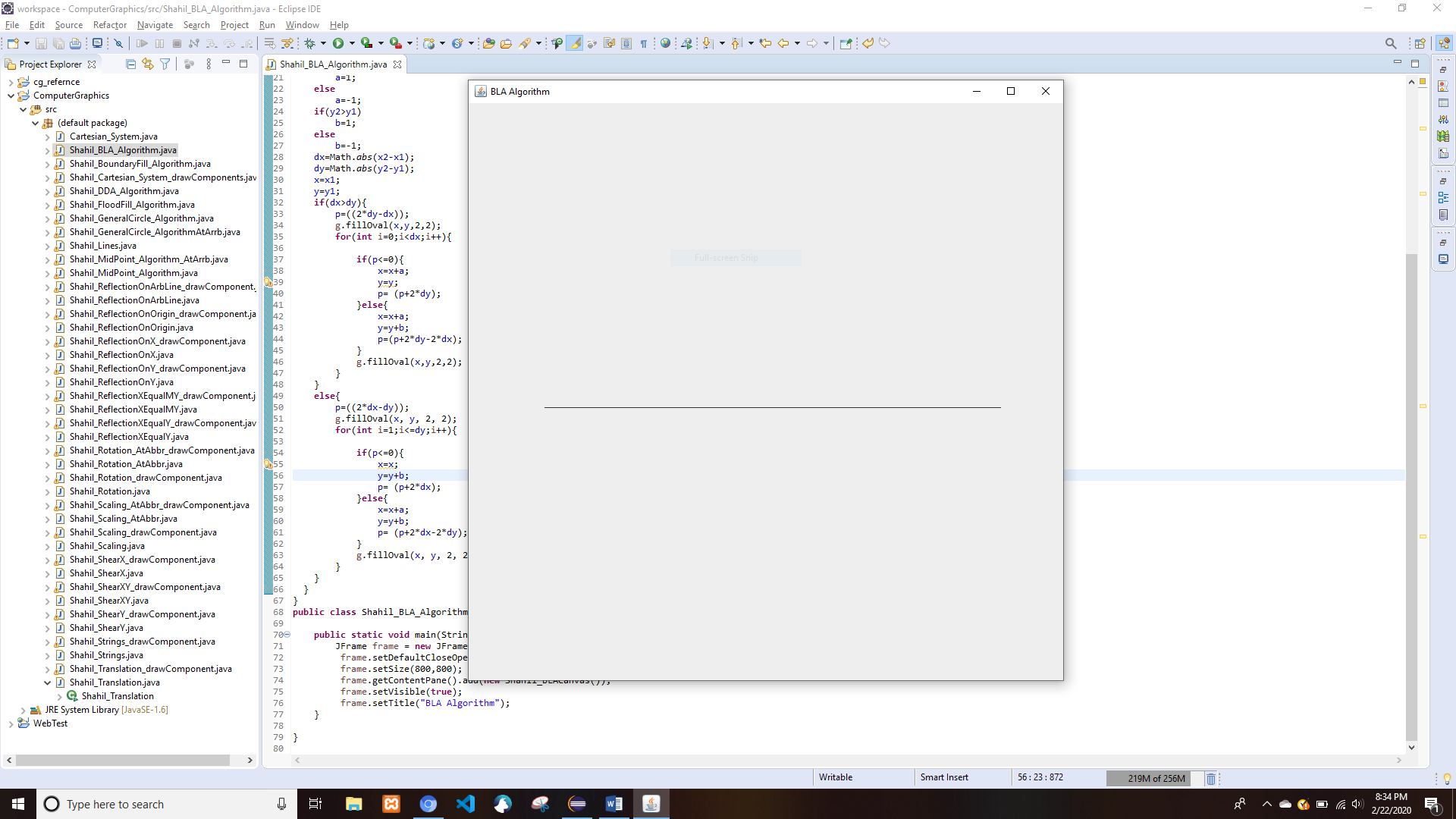
frame.setVisible(**true**);

frame.setTitle("BLA Algorithm");

}

}

* Output



# Digital Differential Analyzer (DDA) Algorithm

* Shahil\_DDA\_Algorithm.java

**import** java.awt.BasicStroke;

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** javax.swing.JComponent;

**import** javax.swing.JFrame;

**class** Shahil\_DDACanvas **extends** JComponent {

**int** x1,x2,y1,y2;

**double** Steps;

**double** dx,dy;

**double** xINC,yINC;

**double** x,y;

**int** xx,yy;

**public** **void** paint(Graphics g) {

Graphics2D g2d = (Graphics2D) g;

x1=100;

y1=400;

x2=700;

y2=400;

//DX VALUES

dx=x2-x1;

dy=y2-y1;

//CHECK

**if**(Math.*abs*(dx)>Math.*abs*(dy)){

Steps=Math.*abs*(dx);

}**else**{

Steps= Math.*abs*(dy);

}

xINC=dx/Steps;

yINC=dy/Steps;

x=x1;

y=y1;

System.***out***.println(Steps);

**for**(**int** i=0;i<Steps;i++){

x=x+xINC;

y=y+yINC;

xx=(**int**)Math.*round*(x);

yy=(**int**)Math.*round*(y);

System.***out***.println(x+" "+y);

g2d.fillOval(xx, yy, 2, 2);

}

}

}

**public** **class** Shahil\_DDA\_Algorithm {

**public** **static** **void** main(String[] args) {

JFrame frame = **new** JFrame();

frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

frame.setSize(800,800);

frame.getContentPane().add(**new** Shahil\_DDACanvas());

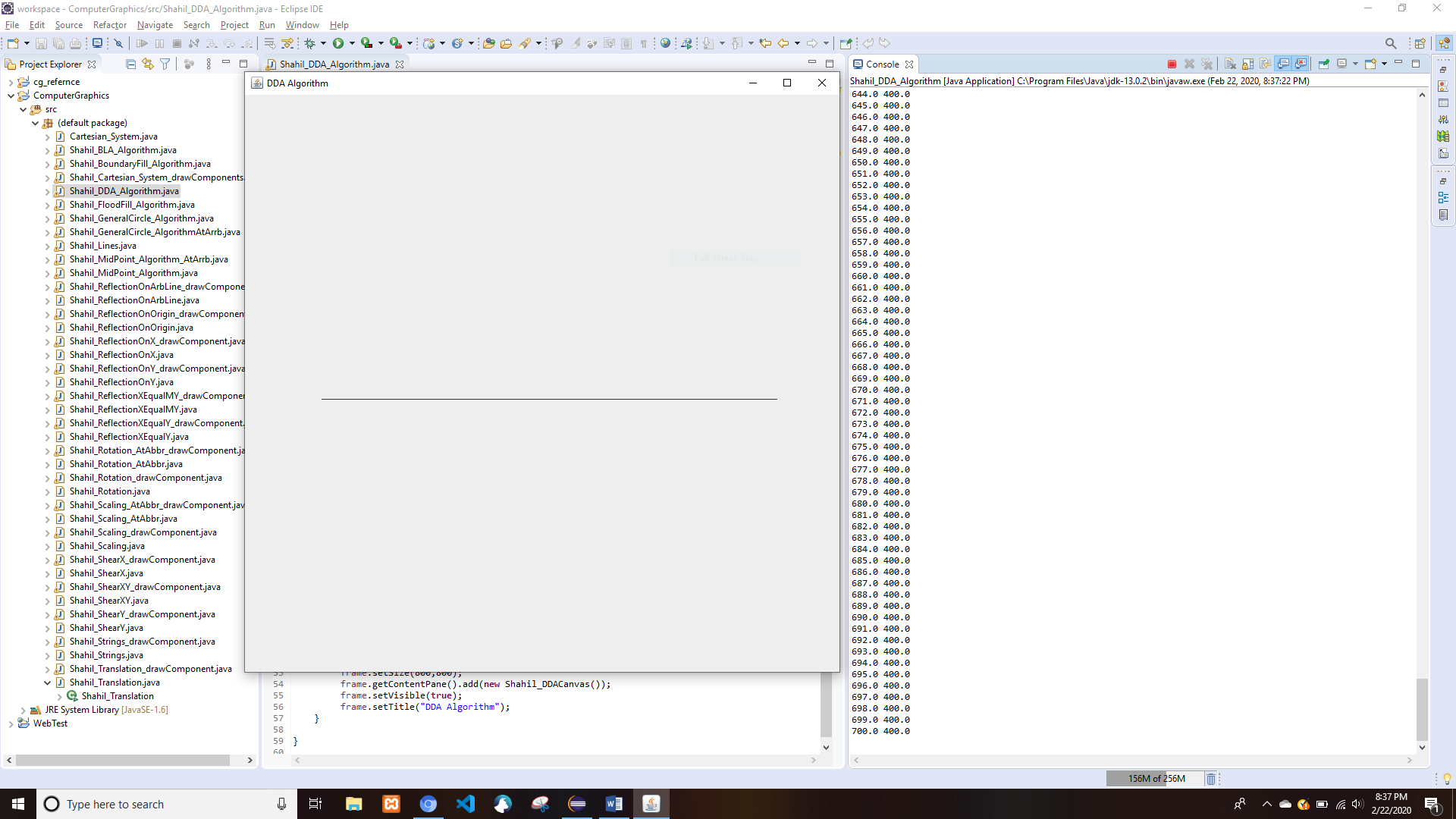
frame.setVisible(**true**);

frame.setTitle("DDA Algorithm");

}

}

* Output



# General Circle Drawing Algorithm

## About Origin

* Shahil\_GeneralCircle\_Algorithm.java

**import** java.awt.Color;

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** java.awt.geom.Line2D;

**import** java.awt.geom.Point2D;

**import** javax.swing.JComponent;

**import** javax.swing.JFrame;

**class** Shahil\_GeneralCircleCanvas **extends** JComponent {

**public** **void** paint(Graphics g) {

Graphics2D g2d = (Graphics2D) g;

//Making the x and y axes

Line2D.Double x\_line = **new** Line2D.Double(**new** Point2D.Double(0,400), **new** Point2D.Double(800,400));

Line2D.Double y\_line = **new** Line2D.Double(**new** Point2D.Double(400, 0), **new** Point2D.Double(400, 800));

g2d.draw(x\_line);

g2d.draw(y\_line);

g2d.translate(400, 400);//translating to the middle

**int** xc,yc,r;

xc=0;

yc=0;

r=200;

**int** x=0;

**int** y=r;

**int** p=3-2\*r;

**do** {

**if**(p<0)

p=p+4\*x+6;

**else**{

p=p+4\*(x-y)+10;

y=y-1;

}

x=x+1;

g2d.drawLine(xc+x,yc+y,xc+x,yc+y);

g2d.drawLine(x+xc,yc-y,xc+x,yc-y);

g2d.drawLine(xc-x,yc+y,xc-x,yc+y);

g2d.drawLine(xc-x,yc-y,xc-x,yc-y);

g2d.drawLine(xc+y,yc+x,xc+y,yc+x);

g2d.drawLine(xc+y,yc-x,xc+y,yc-x);

g2d.drawLine(xc-y,yc+x,xc-y,yc+x);

g2d.drawLine(xc-y,yc-x,xc-y,yc-x);

}**while**(x<y);

}

}

**public** **class** Shahil\_GeneralCircle\_Algorithm {

**public** **static** **void** main(String[] args) {

JFrame frame = **new** JFrame();

frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

frame.setSize(800,800);

frame.getContentPane().add(**new** Shahil\_GeneralCircleCanvas());

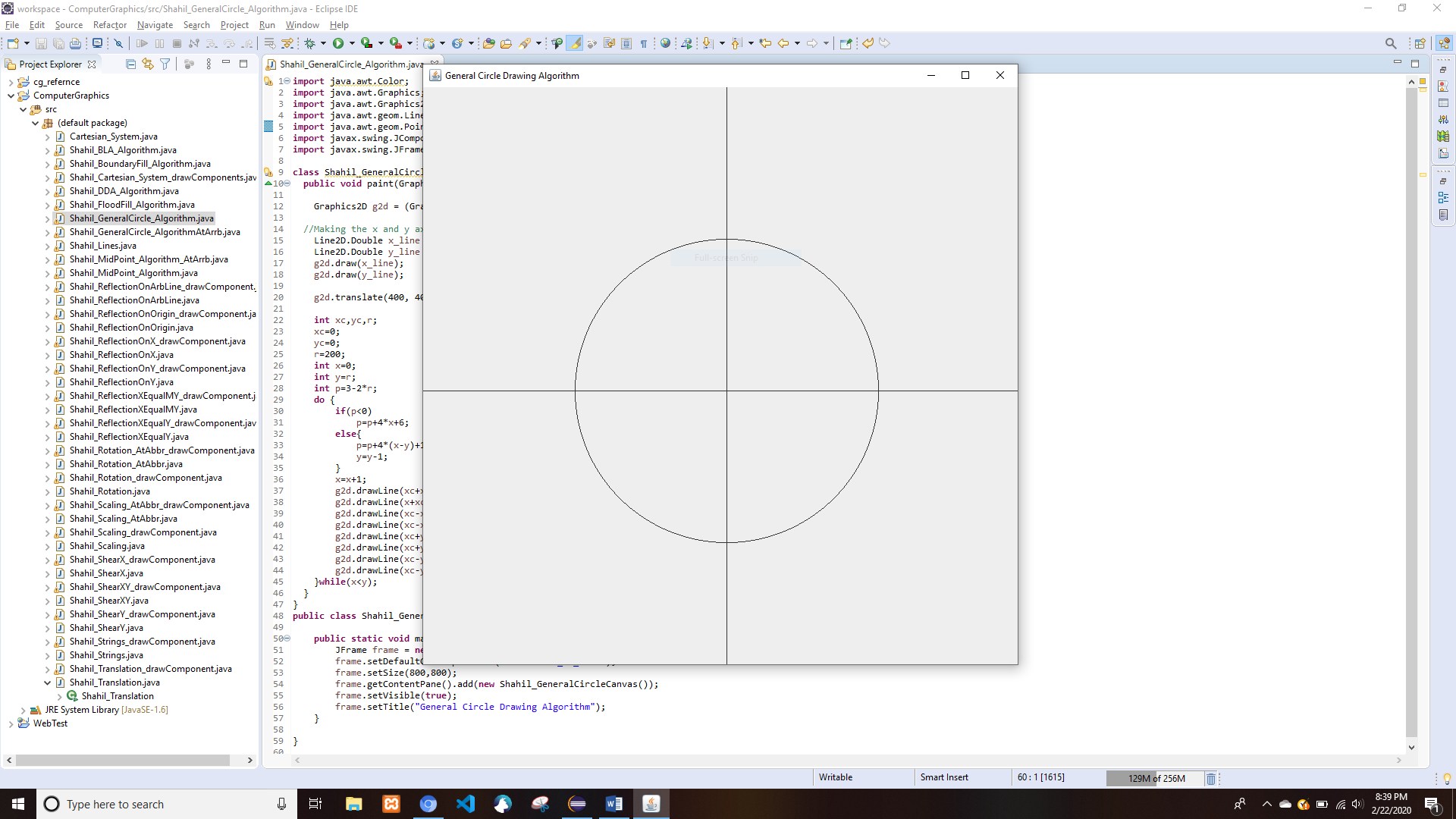
frame.setVisible(**true**);

frame.setTitle("General Circle Drawing Algorithm");

}

}

* Output



## At an Arbitrary Point

* Shahil\_GeneralCircle\_AlgorithmAtArrb.java

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** java.awt.geom.Line2D;

**import** java.awt.geom.Point2D;

**import** javax.swing.JComponent;

**import** javax.swing.JFrame;

**class** Shahil\_GeneralCircleCanvasAtArrb **extends** JComponent {

**public** **void** paint(Graphics g) {

Graphics2D g2d = (Graphics2D) g;

//Making the x and y axes

Line2D.Double x\_line = **new** Line2D.Double(**new** Point2D.Double(0,400), **new** Point2D.Double(800,400));

Line2D.Double y\_line = **new** Line2D.Double(**new** Point2D.Double(400, 0), **new** Point2D.Double(400, 800));

g2d.draw(x\_line);

g2d.draw(y\_line);

g2d.translate(400, 400);//translating to the middle

**int** xc,yc,r,h,k;

xc=0;

yc=0;

r=200;

**int** x=0;

**int** y=r;

h=50;//Arbitrary point x

k=50;//Arbitrary point y

**int** p=3-2\*r;

**do** {

**if**(p<0)

p=p+4\*x+6;

**else**{

p=p+4\*(x-y)+10;

y=y-1;

}

x=x+1;

g2d.drawLine(xc+x+h,yc+y+k,xc+x+h,yc+y+k);

g2d.drawLine(x+xc+h,yc-y+k,xc+x+h,yc-y+k);

g2d.drawLine(xc-x+h,yc+y+k,xc-x+h,yc+y+k);

g2d.drawLine(xc-x+h,yc-y+k,xc-x+h,yc-y+k);

g2d.drawLine(xc+y+h,yc+x+k,xc+y+h,yc+x+k);

g2d.drawLine(xc+y+h,yc-x+k,xc+y+h,yc-x+k);

g2d.drawLine(xc-y+h,yc+x+k,xc-y+h,yc+x+k);

g2d.drawLine(xc-y+h,yc-x+k,xc-y+h,yc-x+k);

}**while**(x<y);

}

}

**public** **class** Shahil\_GeneralCircle\_AlgorithmAtArrb {

**public** **static** **void** main(String[] args) {

JFrame frame = **new** JFrame();

frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

frame.setSize(800,800);

frame.getContentPane().add(**new** Shahil\_GeneralCircleCanvasAtArrb());

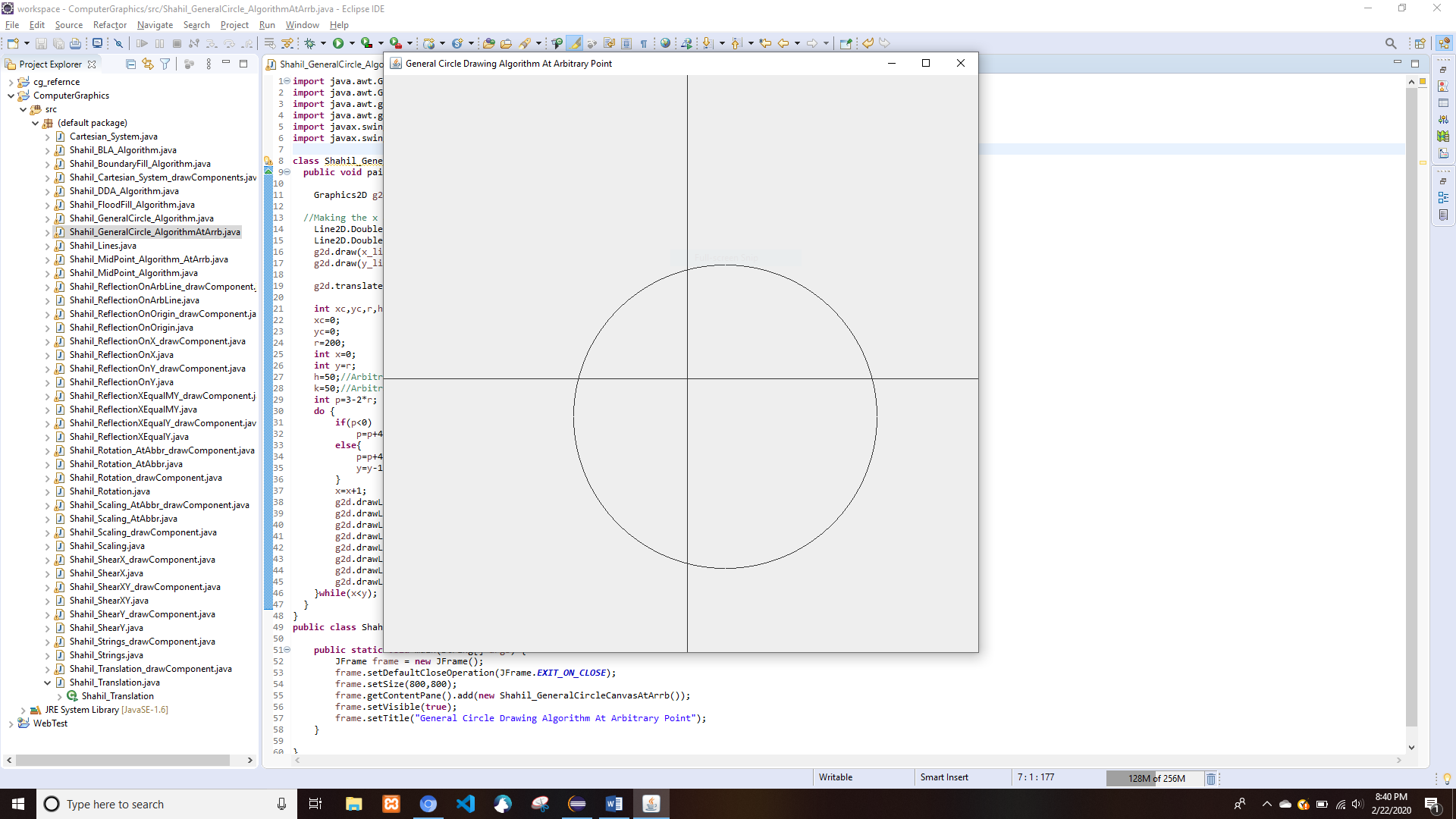
frame.setVisible(**true**);

frame.setTitle("General Circle Drawing Algorithm At Arbitrary Point");

}

}

* Output



# Mid-Point Circle Drawing Algorithm

## About Origin

* Shahil\_MidPoint\_Algorithm.java

**import** java.awt.Color;

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** java.awt.geom.Line2D;

**import** java.awt.geom.Point2D;

**import** javax.swing.JComponent;

**import** javax.swing.JFrame;

**class** Shahil\_MidPointCanvas **extends** JComponent {

**int** r,x,y,p;

**public** **void** paint(Graphics g) {

Graphics2D g2d = (Graphics2D) g;

//Making the x and y axes

Line2D.Double x\_line = **new** Line2D.Double(**new** Point2D.Double(0,400), **new** Point2D.Double(800,400));

Line2D.Double y\_line = **new** Line2D.Double(**new** Point2D.Double(400, 0), **new** Point2D.Double(400, 800));

g2d.draw(x\_line);

g2d.draw(y\_line);

g2d.translate(400, 400);//translating to the middle

r=200;//Radius of circle

x=0;

y=r;

p=1-r;

**while**(x<y){

g2d.setColor(Color.***BLACK***);

g2d.fillOval(x, y,1, 2);

g2d.fillOval(x, -y,1, 2);

g2d.fillOval(-x, -y,1, 2);

g2d.fillOval(-x, y,1, 2);

g2d.fillOval(y, x,1, 2);

g2d.fillOval(y, -x,1, 2);

g2d.fillOval(-y, x,1, 2);

g2d.fillOval(-y, -x,1, 2);

**if**(p<0){

x+=1;

p=p+2\*x+1;

}**else**{

x+=1;

y-=1;

p=p+2\*x-2\*y+1;

}

}

}

}

**public** **class** Shahil\_MidPoint\_Algorithm {

**public** **static** **void** main(String[] args) {

JFrame frame = **new** JFrame();

frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

frame.setSize(800,800);

frame.getContentPane().add(**new** Shahil\_MidPointCanvas());

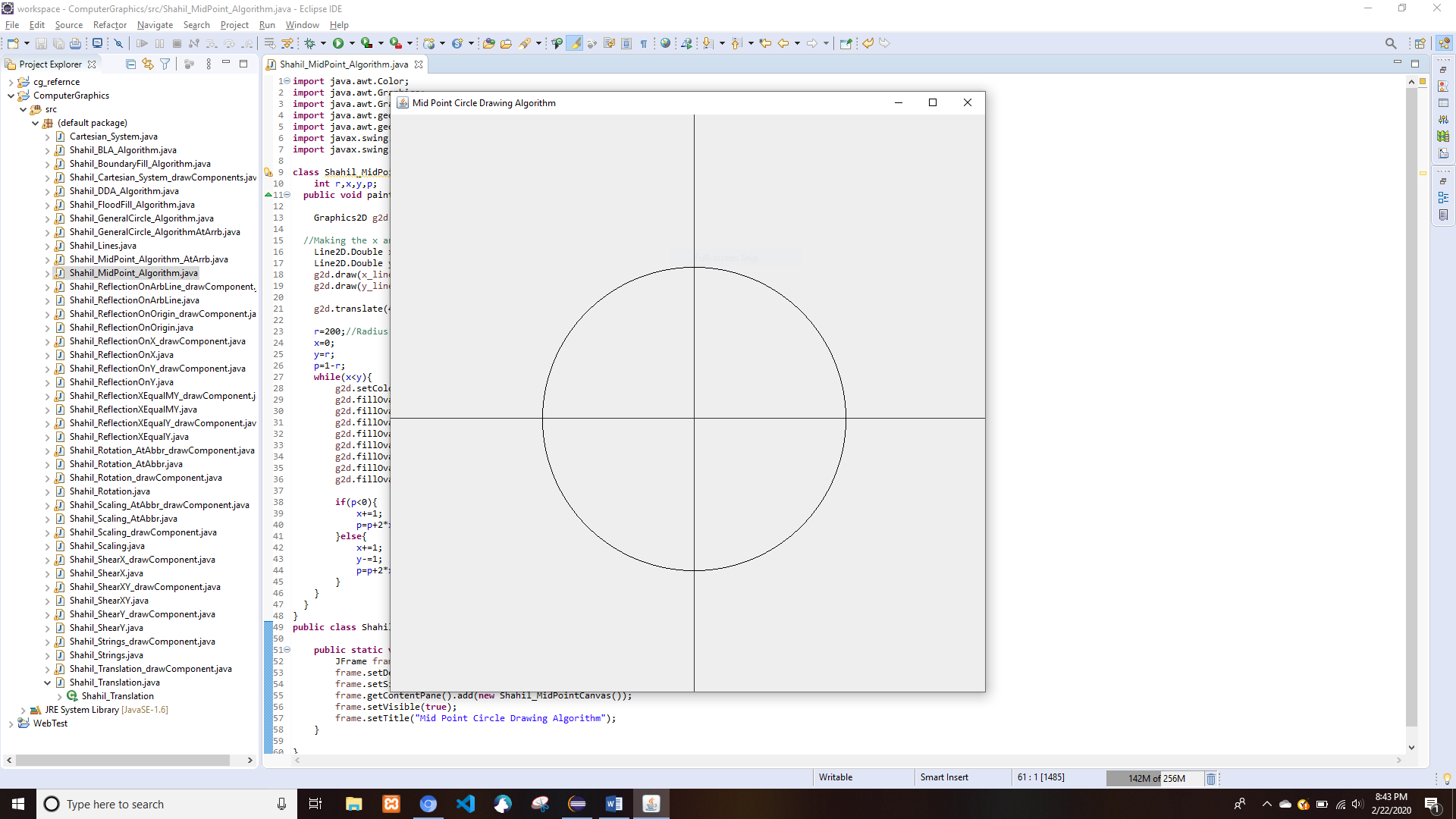
frame.setVisible(**true**);

frame.setTitle("Mid Point Circle Drawing Algorithm");

}

}

* Output



## About an Arbitrary Point

* Shahil\_MidPoint\_Algorithm\_AtArrb.java

**import** java.awt.Color;

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** java.awt.geom.Line2D;

**import** java.awt.geom.Point2D;

**import** javax.swing.JComponent;

**import** javax.swing.JFrame;

**class** Shahil\_MidAtArrbPointCanvas **extends** JComponent {

**int** r,x,y,p,h,k;

**public** **void** paint(Graphics g) {

Graphics2D g2d = (Graphics2D) g;

//Making the x and y axes

Line2D.Double x\_line = **new** Line2D.Double(**new** Point2D.Double(0,400), **new** Point2D.Double(800,400));

Line2D.Double y\_line = **new** Line2D.Double(**new** Point2D.Double(400, 0), **new** Point2D.Double(400, 800));

g2d.draw(x\_line);

g2d.draw(y\_line);

g2d.translate(400, 400);//translating to the middle

r=200;//Radius of circle

x=0;

h=50;//Arbitrary point x

k=50;//Arbitrary point y

y=r;

p=1-r;

**while**(x<y){

g2d.setColor(Color.***BLACK***);

g2d.fillOval(x+h, y+k,1, 2);

g2d.fillOval(x+h, -y+k,1, 2);

g2d.fillOval(-x+h, -y+k,1, 2);

g2d.fillOval(-x+h, y+k,1, 2);

g2d.fillOval(y+h, x+k,1, 2);

g2d.fillOval(y+h, -x+k,1, 2);

g2d.fillOval(-y+h, x+k,1, 2);

g2d.fillOval(-y+h, -x+k,1, 2);

**if**(p<0){

x+=1;

p=p+2\*x+1;

}**else**{

x+=1;

y-=1;

p=p+2\*x-2\*y+1;

}

}

}

}

**public** **class** Shahil\_MidPoint\_Algorithm\_AtArrb {

**public** **static** **void** main(String[] args) {

JFrame frame = **new** JFrame();

frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

frame.setSize(800,800);

frame.getContentPane().add(**new** Shahil\_MidAtArrbPointCanvas());

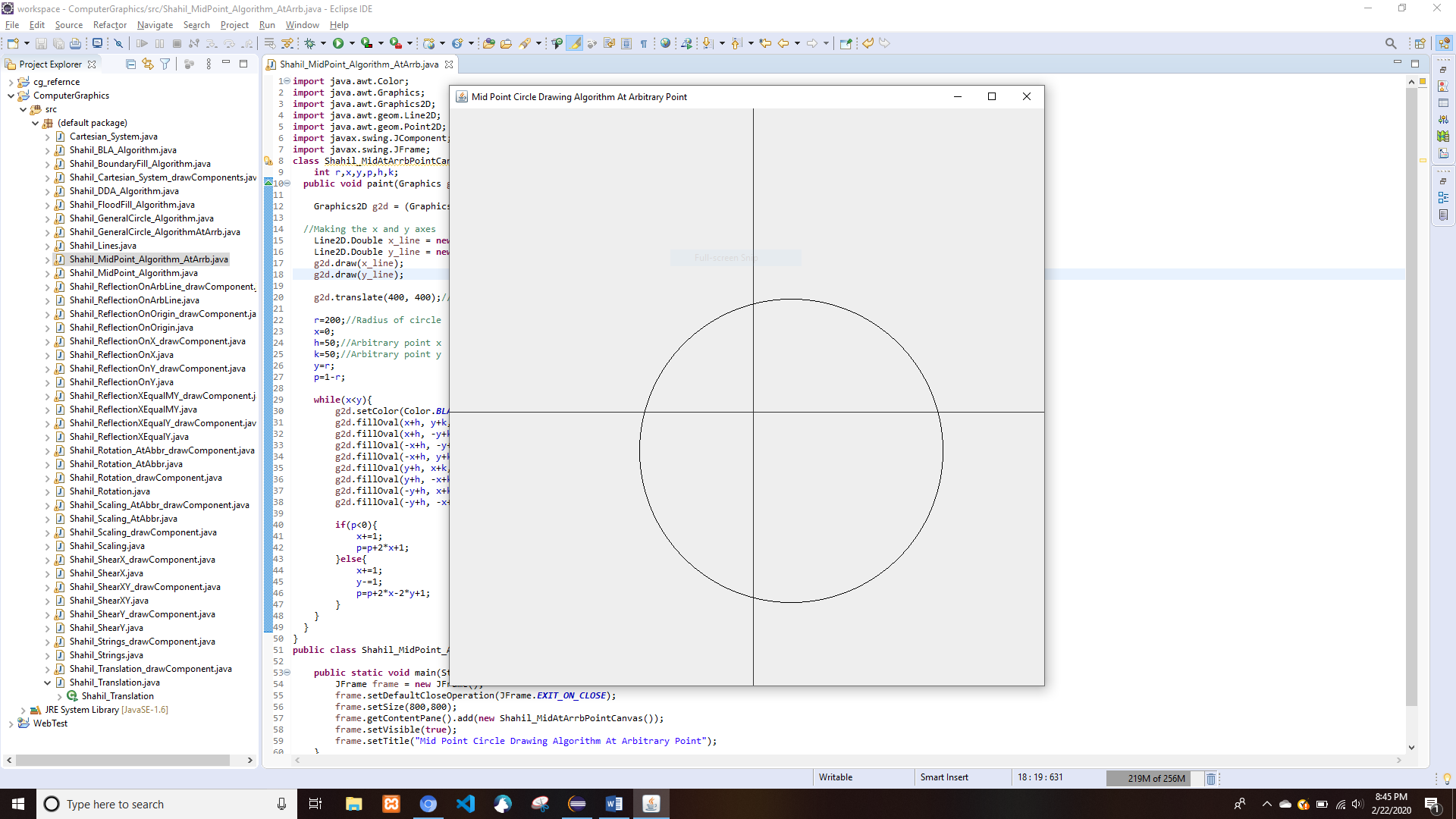
frame.setVisible(**true**);

frame.setTitle("Mid Point Circle Drawing Algorithm At Arbitrary Point");

}

}

* Output



# Boundary Fill Algorithm

* Shahil\_BoundaryFill\_Algorithm.java

**import** java.awt.Color;

**import** java.awt.Dimension;

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** java.awt.event.MouseAdapter;

**import** java.awt.event.MouseEvent;

**import** java.awt.image.BufferedImage;

**import** javax.swing.JFrame;

**import** javax.swing.JPanel;

**public** **class** Shahil\_BoundaryFill\_Algorithm **extends** JPanel{

**final** **int** boundaryColor = Color.***RED***.getRGB();

**final** **int** fillColor = Color.***GREEN***.getRGB();

BufferedImage img;

Graphics2D g2;

Shahil\_BoundaryFill\_Algorithm(){

img = **new** BufferedImage(800,800,BufferedImage.***TYPE\_INT\_RGB***);

setPreferredSize(**new** Dimension(800,800));

setMinimumSize(getPreferredSize());

g2 = img.createGraphics();

g2.setColor(Color.***green***);

g2.clearRect(0, 0, 800, 800);

g2.drawRect(350 , 300, 100, 100);

addMouseListener(**new** MouseAdapter() {

**public** **void** mouseClicked(MouseEvent e) {

System.***out***.println(e.getX()+" "+e.getY());

boundFill(e.getX(), e.getY(), img.getRGB(e.getX(), e.getY()));

}

});

}

**public** **void** paintComponent(Graphics g){

g.drawImage(img, 0, 0, **null**);

}

**public** **void** boundFill(**int** seedX,**int** seedY,**int** rgb)

{

**if**(img.getRGB(seedX, seedY)!=boundaryColor && img.getRGB(seedX, seedY)!=fillColor)

{

img.setRGB(seedX,seedY,fillColor);

update(getGraphics());

boundFill(seedX,seedY-1,rgb);

boundFill(seedX,seedY+1,rgb);

boundFill(seedX-1,seedY,rgb);

boundFill(seedX+1,seedY,rgb);

}

}

**public** **static** **void** main(String args[]){

JFrame f = **new** JFrame("Boundary fill algorithm");

f.setVisible(**true**);

f.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

Shahil\_BoundaryFill\_Algorithm fill = **new** Shahil\_BoundaryFill\_Algorithm();

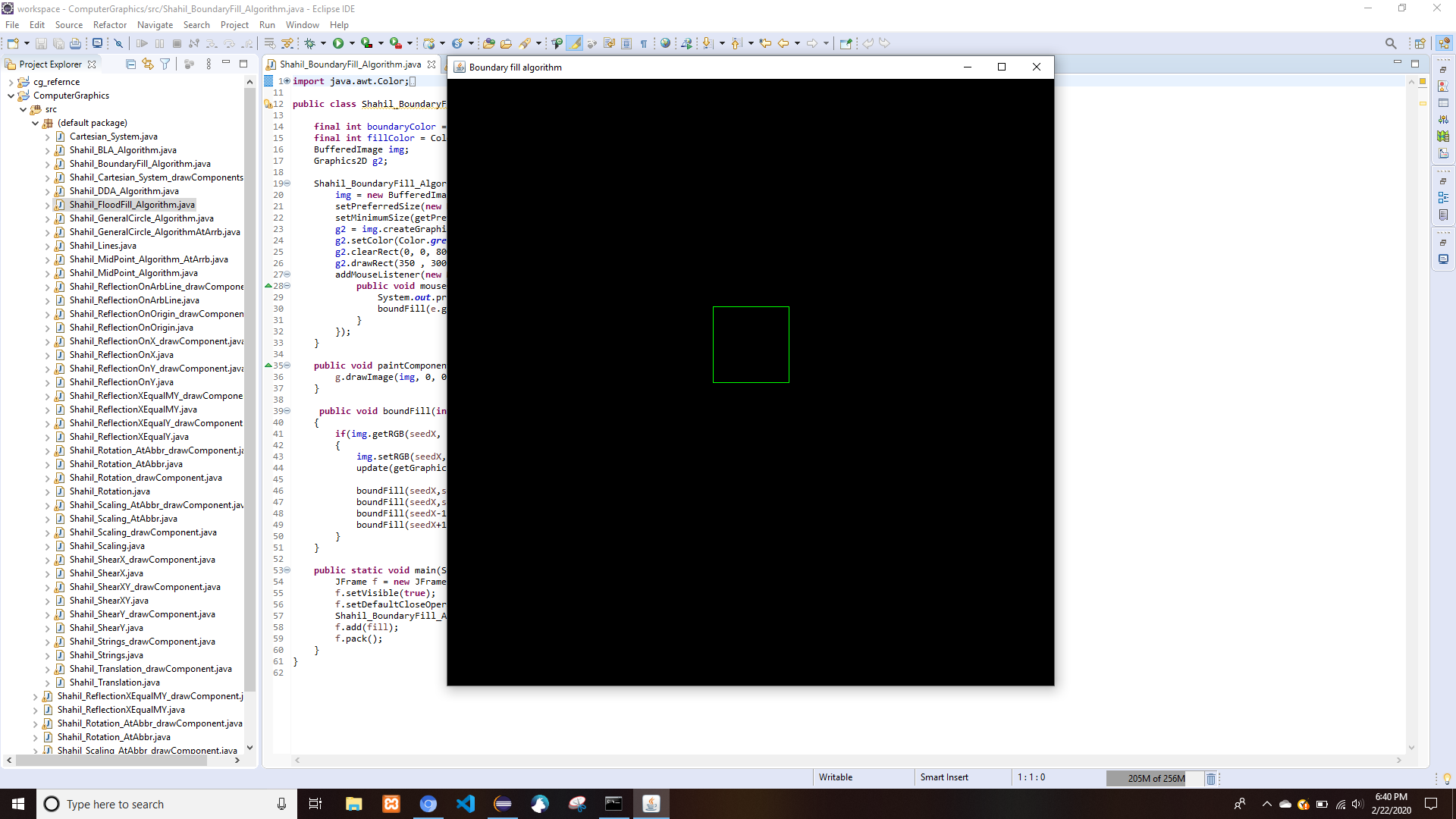
f.add(fill);

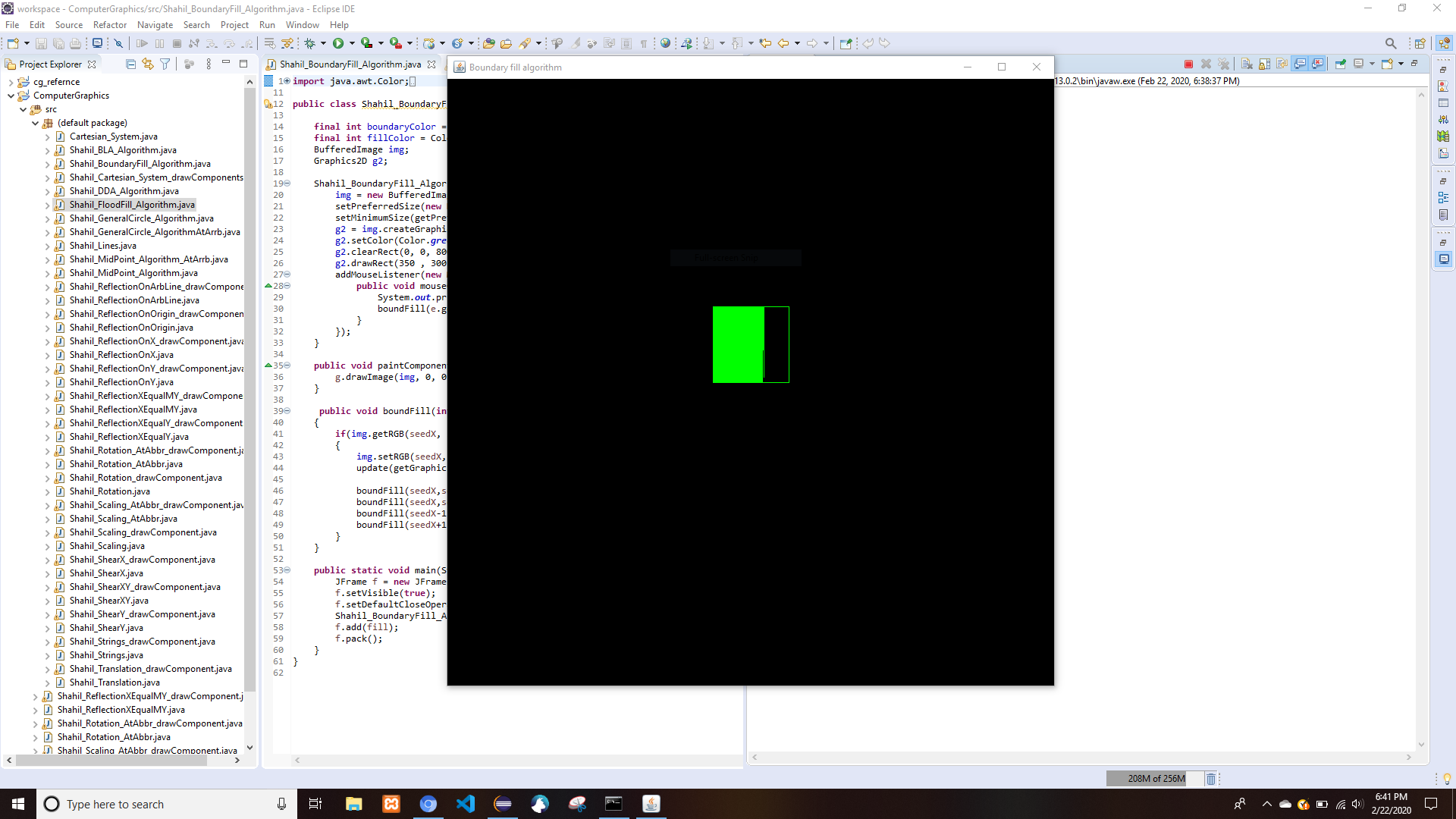
f.pack();

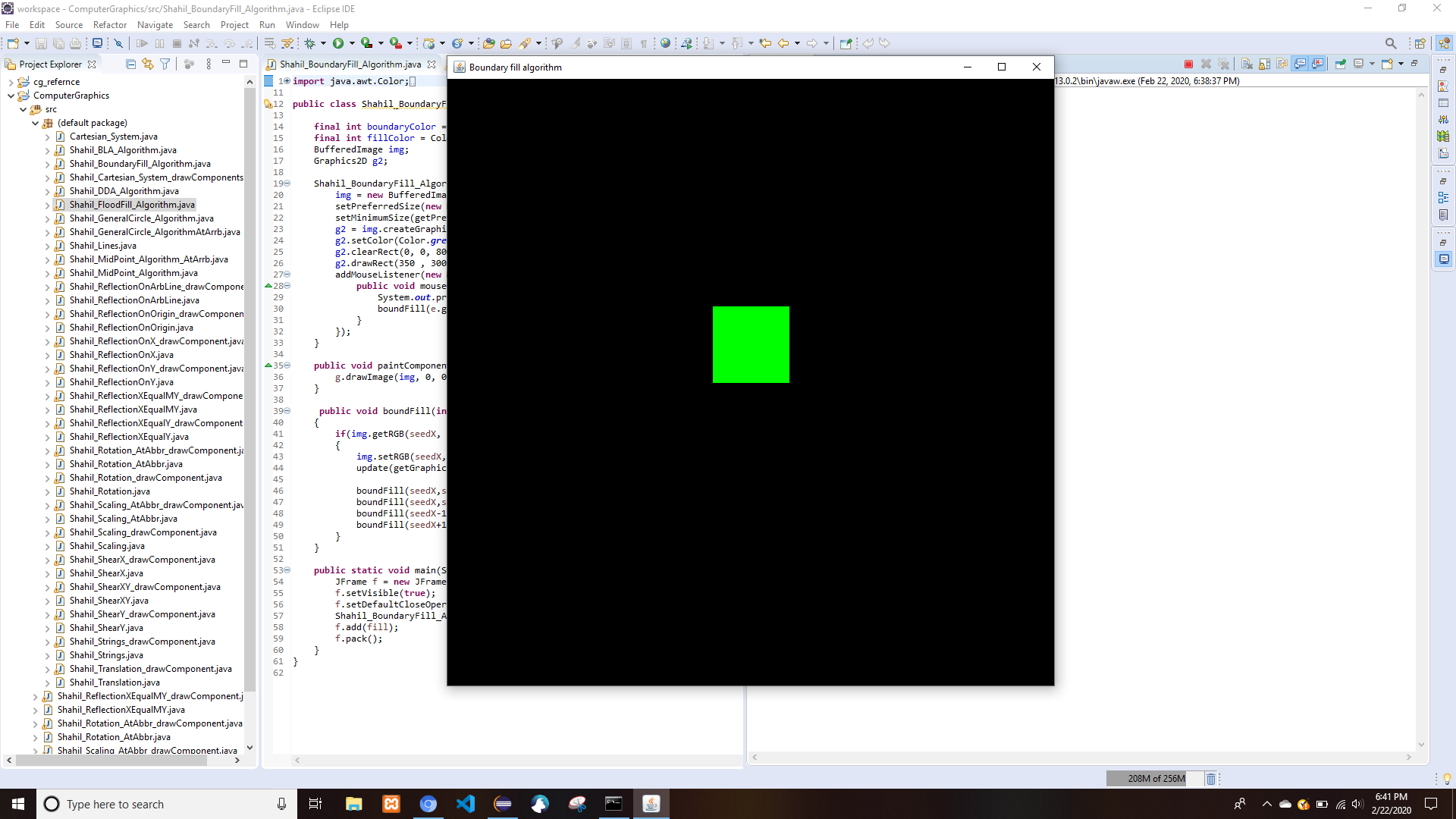
}

}

* Output







# Flood Fill Algorithm

* Shahil\_FloodFill\_Algorithm.java

**import** java.awt.Color;

**import** java.awt.Dimension;

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**import** java.awt.event.MouseAdapter;

**import** java.awt.event.MouseEvent;

**import** java.awt.image.BufferedImage;

**import** javax.swing.JFrame;

**import** javax.swing.JPanel;

**public** **class** Shahil\_FloodFill\_Algorithm **extends** JPanel{

**final** **int** targetColor = Color.***BLUE***.getRGB();

BufferedImage img;

Graphics2D g2;

Shahil\_FloodFill\_Algorithm(){

img = **new** BufferedImage(800,800,BufferedImage.***TYPE\_INT\_RGB***);

setPreferredSize(**new** Dimension(800,800));

setMinimumSize(getPreferredSize());

g2 = img.createGraphics();

g2.setBackground(Color.***WHITE***);

g2.setColor(Color.***blue***);

g2.clearRect(0, 0, 800, 800);

g2.setColor(Color.***BLUE***);

g2.fillRect(350 , 300, 100, 100);

g2.setColor(Color.***GREEN***);

g2.fillRect(375 , 325, 50, 50);

addMouseListener(**new** MouseAdapter() {

**public** **void** mouseClicked(MouseEvent e) {

floodFill(e.getX(), e.getY(), img.getRGB(e.getX(), e.getY()));

}

});

}

**public** **void** paintComponent(Graphics g){

g.drawImage(img, 0, 0, **null**);

}

**public** **void** floodFill(**int** seedX,**int** seedY,**int** rgb)

{

**if**(img.getRGB(seedX, seedY)==targetColor)

{

img.setRGB(seedX,seedY,Color.***red***.getRGB());

update(getGraphics());

floodFill(seedX,seedY-1,rgb);

floodFill(seedX,seedY+1,rgb);

floodFill(seedX-1,seedY,rgb);

floodFill(seedX+1,seedY,rgb);

}

}

**public** **static** **void** main(String args[]){

JFrame f = **new** JFrame("Flood fill algorithm");

f.setVisible(**true**);

f.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

Shahil\_FloodFill\_Algorithm fill = **new** Shahil\_FloodFill\_Algorithm();

f.add(fill);

f.pack();

}

}

