/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* RectangleIntersectionAgain.java

\* Hunter Damron

\* Purpose: Generates three random rectangles in a 600 x 400 plane with width of 50 to 200 and height of 50 to 100;

\* Generates rectangles which represent the intersections of rectangles including one for the intersection of all three;

\* Outputs all three rectangles for reference

\* Outputs the intersection dimensions, or outputs "No intersections" if that is the case

\* Honor Statement: On my honor, I have neither given nor received any unauthorized help on this assignment

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**import** java.awt.Rectangle; //import for Rectangle class

**import** java.util.Random; //import for Rectangle class

**public** **class** RectangleIntersectionAgain {

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

Random generator = **new** Random(); //Random object to generate random numbers

**int** topx; //Top left corner of Rectangles (reused between Rectangles)

**int** topy; //Top right corner of Rectangles (reused between Rectangles)

**int** width; //Width of Rectangles (reused between Rectangles)

**int** height; //Height of Rectangles (reused between Rectangles)

**boolean** out = **false**; //Boolean to test if anything is outputted to determine if a no intersection is appropriate

Rectangle r1, r2, r3, r12, r23, r31, rall; //All rectangles and intersections

width = generator.nextInt(151) + 50; //Generates width between 50 and 200, inclusive

height = generator.nextInt(51) + 50; //Generates height between 50 and 100, inclusive

topx = generator.nextInt(601 - width); //Generates topx so entire Rectangle is between 0 and 600, inclusive

topy = generator.nextInt(401 - height); //Generates topy so entire Rectangle is between 0 and 600, inclusive

r1 = **new** Rectangle(topx, topy, width, height); //Generates first Rectangle r1

width = generator.nextInt(151) + 50; /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

height = generator.nextInt(51) + 50; /\* Same as above but for \*/

topx = generator.nextInt(601 - width); /\* Rectangle r2 \*/

topy = generator.nextInt(401 - height); /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

r2 = **new** Rectangle(topx, topy, width, height); //Generates second Rectangle r2

width = generator.nextInt(151) + 50; /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

height = generator.nextInt(51) + 50; /\* Same as above but for \*/

topx = generator.nextInt(601 - width); /\* Rectangle r2 \*/

topy = generator.nextInt(401 - height); /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

r3 = **new** Rectangle(topx, topy, width, height); //Generates third Rectangle r3

System.***out***.println("// r1: " + r1 + "\n// r2: " + r2 + "\n// r3: " + r3); //Prints the three Rectangles

**if**(r1.intersects(r2)) { //Continues to create intersection of r1 and r2 if any

r12 = r1.intersection(r2); //Makes Rectangle r12 into intersection of r1 and r2

System.***out***.println("Rectangle r1 intersects r2 at " + r12); //Prints intersection

out = **true**; //Sets out to true because there is an intersection

}

**if**(r2.intersects(r3)) { //Continues to create intersection of r2 and r3 if any

r23 = r2.intersection(r3); //Makes Rectangle r23 into intersection of r2 and r3

System.***out***.println("Rectangle r2 intersects r3 at " + r23); //Prints intersection

out = **true**; //Sets out to true because there is an intersection

}

**if**(r3.intersects(r1)) { //Continues to create intersection of r3 and r1

r31 = r3.intersection(r1); //Makes Rectangle r31 into intersection of r3 and r1

System.***out***.println("Rectangle r3 intersects r1 at " + r31); //Prints intersection

out = **true**; //Sets out to true because there is an intersection

**if**(r31.intersects(r2)) { //Continues to create intersection of r31 and r2 which is the intersection of all three if any

rall = r31.intersection(r2); //Makes Rectangle rall into intersection of all r31 and r2

System.***out***.println("All Rectangles intersect at " + rall); //Prints intersection

}

}

**if**(!out) { //Continues to print no intersection if out has not been set to true (meaning no intersection is present)

System.***out***.println("No intersections were detected"); //Prints no intersection

}

}

}

Outputs

With no intersections:

// r1: java.awt.Rectangle[x=389,y=52,width=74,height=86]

// r2: java.awt.Rectangle[x=35,y=72,width=61,height=73]

// r3: java.awt.Rectangle[x=132,y=248,width=97,height=50]

No intersections were detected

With one intersection:

// r1: java.awt.Rectangle[x=484,y=279,width=95,height=62]

// r2: java.awt.Rectangle[x=362,y=266,width=146,height=72]

// r3: java.awt.Rectangle[x=333,y=136,width=137,height=68]

Rectangle r1 intersects r2 at java.awt.Rectangle[x=484,y=279,width=24,height=59]

With two intersections:

// r1: java.awt.Rectangle[x=162,y=149,width=147,height=50]

// r2: java.awt.Rectangle[x=200,y=39,width=199,height=88]

// r3: java.awt.Rectangle[x=112,y=75,width=159,height=86]

Rectangle r2 intersects r3 at java.awt.Rectangle[x=200,y=75,width=71,height=52]

Rectangle r3 intersects r1 at java.awt.Rectangle[x=162,y=149,width=109,height=12]

With three intersections:

// r1: java.awt.Rectangle[x=363,y=102,width=162,height=91]

// r2: java.awt.Rectangle[x=195,y=91,width=190,height=74]

// r3: java.awt.Rectangle[x=287,y=16,width=93,height=93]

Rectangle r1 intersects r2 at java.awt.Rectangle[x=363,y=102,width=22,height=63]

Rectangle r2 intersects r3 at java.awt.Rectangle[x=287,y=91,width=93,height=18]

Rectangle r3 intersects r1 at java.awt.Rectangle[x=363,y=102,width=17,height=7]

All Rectangles intersect at java.awt.Rectangle[x=363,y=102,width=17,height=7]