

ข้อสอบปลายภาค

รหัสวิชา 02739214 Principles of Object Oriented Programming

สอบวันที่ 2 เมษายน พ.ศ. 2565

คำชี้แจง

1. ข้อสอบโปรแกรมมิ่งมี 4 ข้อคะแนนรวม 45 คะแนน
2. ส่งข้อสอบทาง google classroom
3. ให้ทำการ zip รวมไฟล์ทุกโปรเจกต์รวมเป็นไฟล์เดียวและทำการตั้งชื่อเป็น
<รหัสนิสิต>_<ชื่อภาษาอังกฤษ>_<นามสกุลภาษาอังกฤษ>.zip
เช่น 44010557_Surdej_Intagorn.zip
4. อนุญาตให้เปิดเอกสารและค้นหาข้อมูลในอินเทอร์เน็ตในระหว่างทำข้อสอบได้
อย่างไรก็ตามการ ลอกคำตอบของผู้เข้าสอบคนอื่นถือเป็นการทุจริต
หากตรวจสอบแล้ว มีข้อสงสัยเกี่ยวกับการทุจริตเรื่องการลอกคำตอบของผู้เข้าสอบคนอื่นมาส่ง
จะมีการเรียกเข้ามาพบเพื่อให้ทำการทดสอบเพิ่มเติม

ชื่อ-นามสกุล..... รหัสนักศึกษา..... หมู่เรียน..... เลขที่.....

ข้อ1 ตั้งชื่อโปรเจคเป็น Q001final (8 คะแนน)

เขียน JAVA class ชื่อ Product จาก class diagram ด้านล่าง ให้ทำการ encapsulation/ สร้าง constructor แบบรับค่า เป็น attributes ทั้งหมด overriding equals/toString/hashCode จากคลาส Object เมธอด toString ที่ override ให้ ทำการแสดงค่า attributes ทั้งหมด

หมายเหตุ getter/setter, toString, equals, hashCode ไม่ได้อยู่ใน class diagram

Product
-productID:int -productName:String -categoryID:int -unit:String -price:double
+Product(productID:int, productName:String, categoryID:int, unit:String, price:double)

ให้สร้าง ProductTest.java เพื่อทำการสร้าง object ของคลาส Product ขึ้นมา 4 ตัวโดยทำการกำหนดค่าของ attributes ผ่านทาง constructors ดังตารางด้านล่าง

productID	productName	categoryID	unit	price
1	Chais	1	boxes	18
2	Chang	1	bottles	19
3	Aniseed Syrup	1	bottles	10
4	Chef Anton's Cajun Seasoning	2	jars	22

จากนั้นให้ทำการแสดงค่าของออบเจกต์ Product ทั้ง 4 ตัวด้วยคำสั่ง System.out.println

ข้อ2 ตั้งชื่อโปรเจกเป็น Q002final (12 คะแนน)

การสร้างอะเรย์ของออบเจกต์ในจาวา

อ่านไฟล์ `pokedex.json` และทำการสร้างออบเจกต์ `Pokemon` โดยคลาส `Pokemon`

มีรายละเอียดตาม `class diagram` ด้านล่าง ให้ทำการ `encapsulation`/ สร้าง `constructor` แบบรับค่าเป็น `attributes`

ทั้งหมด `overriding equals/toString/hashCode` จากคลาส `Object` เมธอด `toString` ที่ `override` ให้ทำการแสดงค่า `attributes` ทั้งหมด

หมายเหตุ `getter/setter, toString, equals, hashCode` ไม่ได้อยู่ใน `class diagram`

Pokemon
<code>-id : Integer</code> <code>-name_english : String</code> <code>-attack: Double</code> <code>-defense : Double</code> <code>-spAttack : Double</code> <code>-spDefense : Double</code>
<code>+Pokemon(id:int, name_english:String, attack:Double, defense:Double, spAttack:Double, spDefense:Double)</code>

จากนั้นให้ทำการเรียงออบเจกต์ `Pokemon` จากค่า `spAttack` โดยเรียงจากน้อยไปมาก และทำการแสดงผลบนคอนโซลด้วย

คำสั่ง `System.out.println`

ข้อ3 ตั้งชื่อโปรเจกเป็น Q003final (15 คะแนน)

สร้างคลาส Pokemon มีรายละเอียดตาม class diagram ด้านล่าง (เหมือนข้อ 2) ให้ทำการ encapsulation/ สร้าง constructor แบบรับค่าเป็น attributes ทั้งหมด overriding equals/toString/hashCode จากคลาส Object เมธอด toString ที่ override ให้ทำการแสดงค่า attributes ทั้งหมด

หมายเหตุ getter/setter, toString, equals, hashCode ไม่ได้อยู่ใน class diagram

Pokemon
-id : Integer -name_english : String -attack: Double -defense : Double -spAttack : Double -spDefense : Double
+Pokemon(id:int, name_english:String, attack:Double, defense:Double, spAttack:Double, spDefense:Double)

จากนั้นทำการสร้างคลาส PokemonService ดัง class diagram ด้านล่างเพื่อใช้ในการอ่านไฟล์ pokedex.json เพื่อเก็บข้อมูลโปเกมอนใน list และทำการสร้างเมธอดในการจัดการข้อมูลเหล่านั้น

PokemonService
-pokemon_list:ArrayList<Pokemon>
+PokemonService() +sortByNameEnglish(): ArrayList<Pokemon> +sortByAttack(): ArrayList<Pokemon> +sortByDefense(): ArrayList<Pokemon> +searchByNameEnglish(name:String) : ArrayList<Pokemon> +searchBySpAttack(min_spAttack:double, max_spAttack:double) : ArrayList<Pokemon> +insertPokemon(id:int, name_english:String, attack:Double, defense:Double, spAttack:Double, spDefense:Double) +updatePokemon(id:int, name_english:String, attack:Double, defense:Double, spAttack:Double, spDefense:Double) +deletePokemon(id : Integer)

PokemonService จะต้องทำตามคลาสไดอะแกรมนี้เป็นขั้นต่ำ อย่างไรก็ตาม, รายละเอียดของการเขียนโค้ดในเมธอดสามารถทำได้อย่างอิสระ นอกจากนี้ นักศึกษาสามารถสร้างเมธอดและคลาสอื่นๆเพิ่มเติมได้ตามเหมาะสมเช่น สร้างเมธอดชื่อ

+searchById(id:int) : Pokemon เพิ่มเติม หรือทำการสร้างคลาสเพิ่มเติมเช่น PokemonAttackComparator

จากนั้นให้ทำการสร้างหน้า GUI ชื่อ PokemonGUI ที่ทำการเรียกใช้เมธอดจากคลาส PokemonService ที่ป้อนต่างๆ

การสร้าง **GUI** ให้นิสิตแต่ละคน ออกแบบ/เปลี่ยนตำแหน่งของ **UI** ให้ไม่เหมือนกัน
และทำการเปลี่ยนชื่อของ **UI** แต่ละตัวตามชื่อภาษาอังกฤษของนิสิต ดูคำอธิบายต่อด้านล่าง

The screenshot shows a Java Swing window titled 'PokemonGUI'. On the left, there are six text input fields labeled ID, NAME, Attack, Defense, spAttack, and spDefense. Below these fields are three buttons: ADD (labeled 1), UPDATE (labeled 2), and DELETE (labeled 3). To the right of the input fields, there is a search section with a text box labeled 'ค้นหาด้วยชื่อ' (labeled 7) and a 'ค้นหา' button (labeled 8). Below the search section, there are three radio buttons: 'ชื่อ' (labeled 4), 'พลังโจมตี' (labeled 5), and 'พลังป้องกัน' (labeled 6). A 'เรียง' button is also present. At the bottom right, there is a table with four columns labeled Title 1, Title 2, Title 3, and Title 4. The table contains several rows of data, with the first row highlighted in yellow.

GUIหมายเลข	เรียกใช้เมธอด	Function specification
1	insertPokemon	เพิ่มข้อมูล Pokemon เข้าไปใน list
2	updatePokemon	แก้ไขข้อมูล Pokemon เข้าไปใน list
3	deletePokemon	ลบข้อมูล Pokemon
4	sortByNameEnglish()	เรียงข้อมูล Pokemon จาก name และแสดงผลที่ตาราง
5	sortByAttack ()	เรียงข้อมูล Pokemon จาก attack และแสดงผลที่ตาราง
6	sortByDefense():	เรียงข้อมูล Pokemon จาก defense และแสดงผลที่ตาราง
7	searchByNameEnglish	ค้นหาข้อมูล Pokemon จาก name และแสดงผลที่ตาราง
8	searchBySpAttack	ค้นหาข้อมูล Pokemon จาก spAttack และแสดงผลที่ตาราง

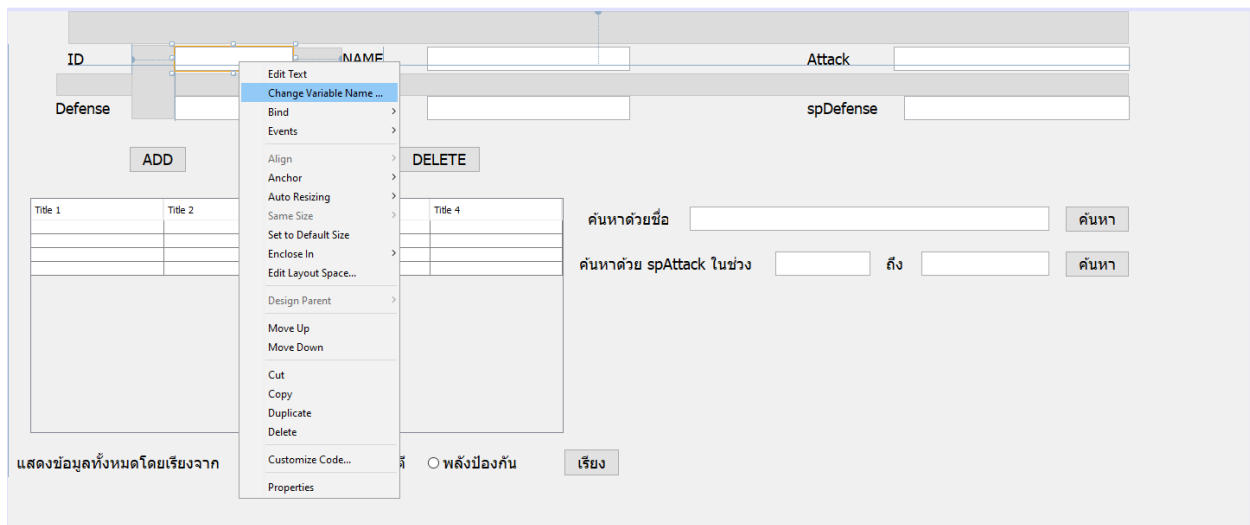
การสร้าง **GUI** ให้นิสิตแต่ละคนวางตำแหน่ง **UI** แตกต่างไปจากรูปตัวอย่าง
ตัวอย่างเช่น โดยให้วางอย่างไรก็ได้แต่ห้ามซ้ำกับ รูปก่อนหน้าและรูปด้านล่าง

จากนั้นให้ทำการเปลี่ยนชื่อตัวแปรของ **jTextField** แต่ละตัวให้ต่างไปจากเดิมอย่างไรก็ได้ โดยแต่ละคนไม่ควรจะซ้ำกัน ตัวอย่างการเปลี่ยนชื่อ **UI** ให้ไม่ซ้ำเช่น

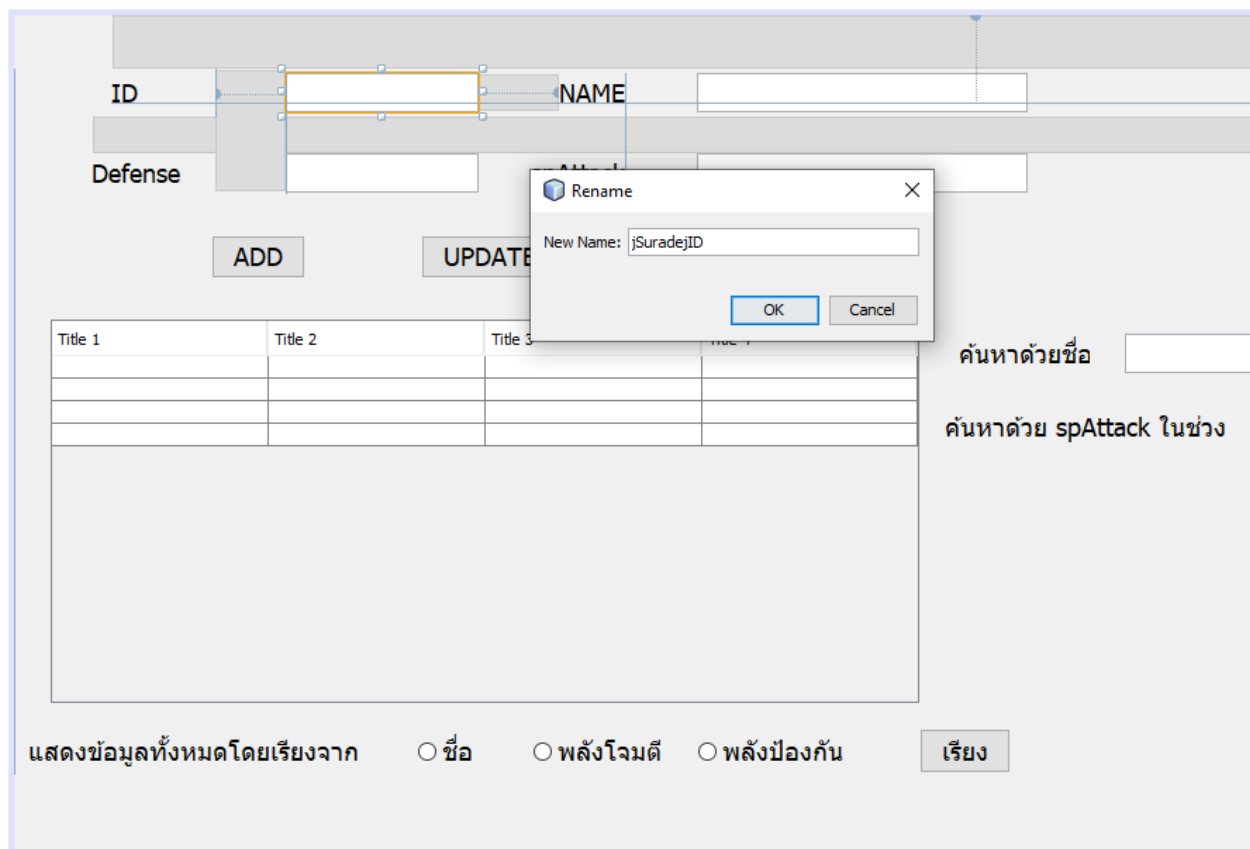
j<ชื่อภาษาอังกฤษของตนเอง><ความหมายของUI>

	ตัวอย่างชื่อเดิม	ตัวอย่าง 1	ตัวอย่าง 2
1	jTextField1	jSuradej1	jSuradejID
2	jTextField2	jSuradej2	jSuradejName
3	jTextField3	jSuradej3	jSuradejAttack
4	jTextField4	jSuradej4	jSuradejDefense
5	jTextField5	jSuradej5	jSuradejSpAttack
6	jTextField6	jSuradej6	jSuradejSpDefense
7	jTextField7	jSuradej7	jSuradejSearchName
8	jTextField8	jSuradej8	jSuradejMinSpAttack
9	jTextField9	jSuradej9	jSuradejMaxSpAttack

การเปลี่ยนชื่อ UI ให้ทำการกดคลิกขวาที่ UI ที่ต้องการเลือก Change Variable Name



จากนั้นเปลี่ยนเป็นชื่อตามต้องการเช่นต้องการเปลี่ยนชื่อเป็น jSuradejID จากนั้นกด OK



ข้อ4 ตั้งชื่อโปรเจคเป็น Q004final (10 คะแนน)

0035 : สามเหลี่ยมใหญ่ (bigtriangle)

ที่มา https://programming.in.th/task/rev2_problem.php?pid=0035

ในขณะที่คุณกำลังสำรวจโบราณสถานของอาณาจักรสุโขทัยโบราณอยู่ คุณได้พบเสาหินโบราณที่หลงเหลือจากการทำลายจำนวนมากตั้งเรียงรายอยู่โดยรอบ เพื่อนของคุณนึกสนุกจึงทำคุณให้หาพื้นที่ทั้งหมดที่ล้อมรอบด้วยเสาหินเหล่านี้ โดยมีรางวัลเป็นขนมแสนอร่อยที่เพื่อนของคุณซื้อมา

ด้วยความที่คุณมีความเป็นนักคณิตศาสตร์อยู่ในตัว คุณทราบดีว่ามันยากเกินไปที่จะหาพื้นที่ดังกล่าวในระยะเวลาอันสั้น คุณจึงขอเพื่อนเปลี่ยนเป็นหาพื้นที่สามเหลี่ยมที่ใหญ่ที่สุดที่มีเสาหินเป็นจุดมุมของสามเหลี่ยมแทน

หมายเหตุ: สำหรับสามเหลี่ยมที่จุดยอดทั้งสามมีพิกัด $(x1, y1)$, $(x2, y2)$ และ $(x3, y3)$ พื้นที่ของสามเหลี่ยมจะมีค่าเท่ากับ $|x1y2 + x2y3 + x3y1 - y1x2 - y2x3 - y3x1|/2$

งานของคุณ

เขียนโปรแกรมรับพิกัดของเสาหินทั้งหมด แล้วคำนวณหาพื้นที่สามเหลี่ยมที่ใหญ่ที่สุดที่มีเสาหินเป็นจุดมุม

ข้อมูลนำเข้า

บรรทัดแรกระบุจำนวนเต็มบวก N ($1 \leq N \leq 100$) แทนจำนวนเสาหินทั้งหมด

จากนั้นอีก N บรรทัดจะระบุพิกัดของเสาหิน กล่าวคือ สำหรับบรรทัดที่ $i+1$ ($1 \leq i \leq N$) ระบุจำนวนเต็มสองจำนวน X_i และ Y_i คั่นด้วยช่องว่าง โดย (X_i, Y_i) คือพิกัดของเสาหินต้นที่ i ($-1,000 \leq X_i, Y_i \leq 1,000$) รับประกันว่าเสาหินสองต้นจะไม่มีพิกัดเดียวกัน

ข้อมูลส่งออก

มีหนึ่งจำนวน เป็นจำนวนจริงแทนพื้นที่ของสามเหลี่ยมที่ใหญ่ที่สุดที่มีเสาหินเป็นจุดมุม แสดงเป็นทศนิยม 3 ตำแหน่ง

ที่มา

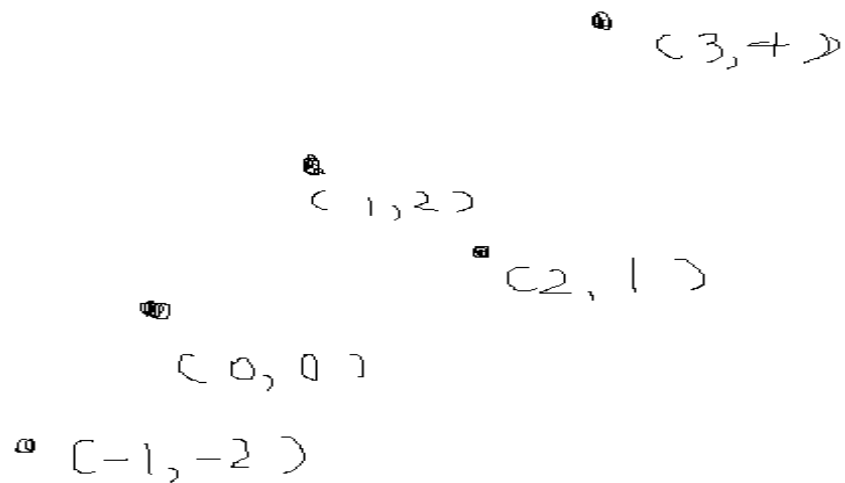
การแข่งขัน IOI Thailand League เดือนกันยายน 2553

โจทย์โดย: ธงชัย วิโรจน์ศักดิ์เสรี

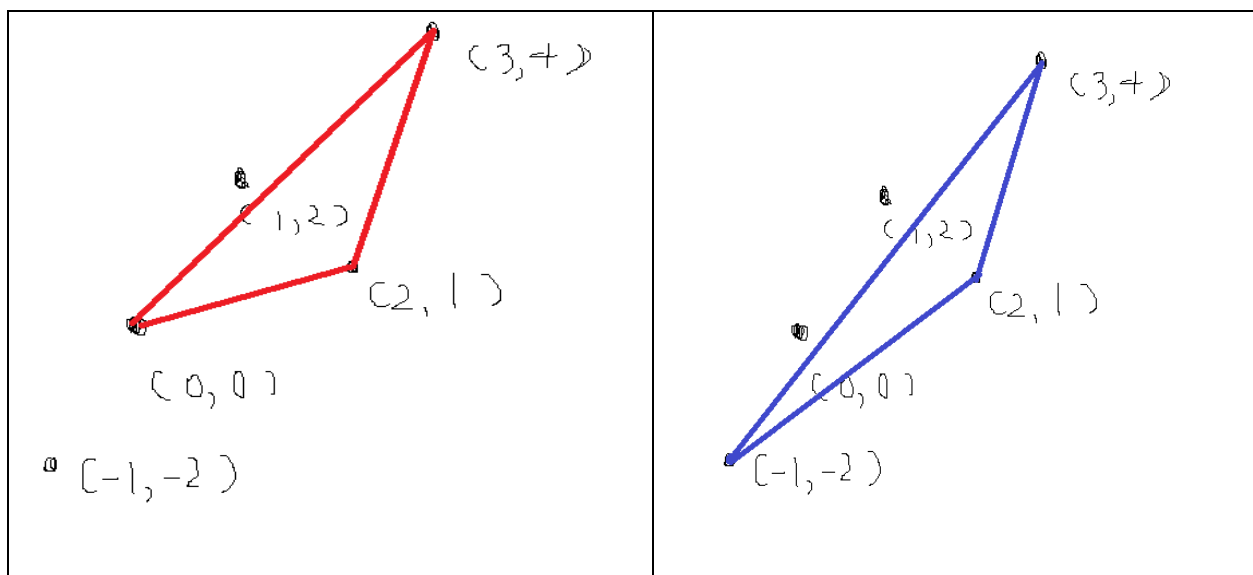
ตัวอย่างข้อมูลนำเข้า	ตัวอย่างข้อมูลส่งออก
5 1 2 2 1 0 0 3 4 -1 -2	3.000
5 1 1 2 2 3 3 1 4 4 1	4.500

ความช่วยเหลือ: ไม่มีคำใบ้สำหรับปัญหานี้

Hint : ในข้อนี้โจทย์สั่งให้ทำการกรอก input เป็น จำนวนจุดในบรรทัดแรก จากนั้นให้วนลูปเพื่อรับพิกัดของจุด ดูตัวอย่างการรับ input ด้านล่างเช่นถ้ากรอกตามตัวอย่างแรกจะได้จะจุดคร่าวๆดังนี้



จากนั้นให้ทำการคำนวณสามเหลี่ยมทั้งหมดที่เป็นไปได้เช่น 5 จุดจะมีสามเหลี่ยม 10 แบบที่ไม่ซ้ำกันด้านล่างเป็นตัวอย่างสามเหลี่ยม จากนั้นหาพื้นที่ของสามเหลี่ยมแต่ละอันและตอบเป็นพื้นที่สามเหลี่ยมที่มากที่สุดใน 10 แบบนี้



ตัวอย่างโค้ดในการรับ input โดยภาษา JAVA ในข้อนี้

```
import java.util.*;

public class Q4 {

    public static void main(String args[]){

        Scanner scan = new Scanner(System.in);

        int N = Integer.parseInt(scan.nextLine());

        int[][] points = new int[N][2] ;

        //รับค่า
        for(int i=0;i<N;i++){

            String line = scan.nextLine();

            String[] data = line.split(" ");

            points[i][0] = Integer.parseInt(data[0]);

            points[i][1] = Integer.parseInt(data[1]);

        }

        //ตัวอย่างการ วาดรูป เพื่อแสดงค่า
        for(int i=0;i<N;i++){

            int x1 = points[i][0];

            int y1 = points[i][1];

            System.out.println("X1 = " + x1 + " Y1 = " + y1);

        }

        //ตัวอย่างการคำนวณพื้นที่สามเหลี่ยมระหว่างจุด (1,2) (2,1) (0,0) และ การใช้คำสั่ง absolute ใน JAVA

        int x1 = points[0][0];
        int y1 = points[0][1];
        int x2 = points[1][0];
        int y2 = points[1][1];
        int x3 = points[2][0];
        int y3 = points[2][1];

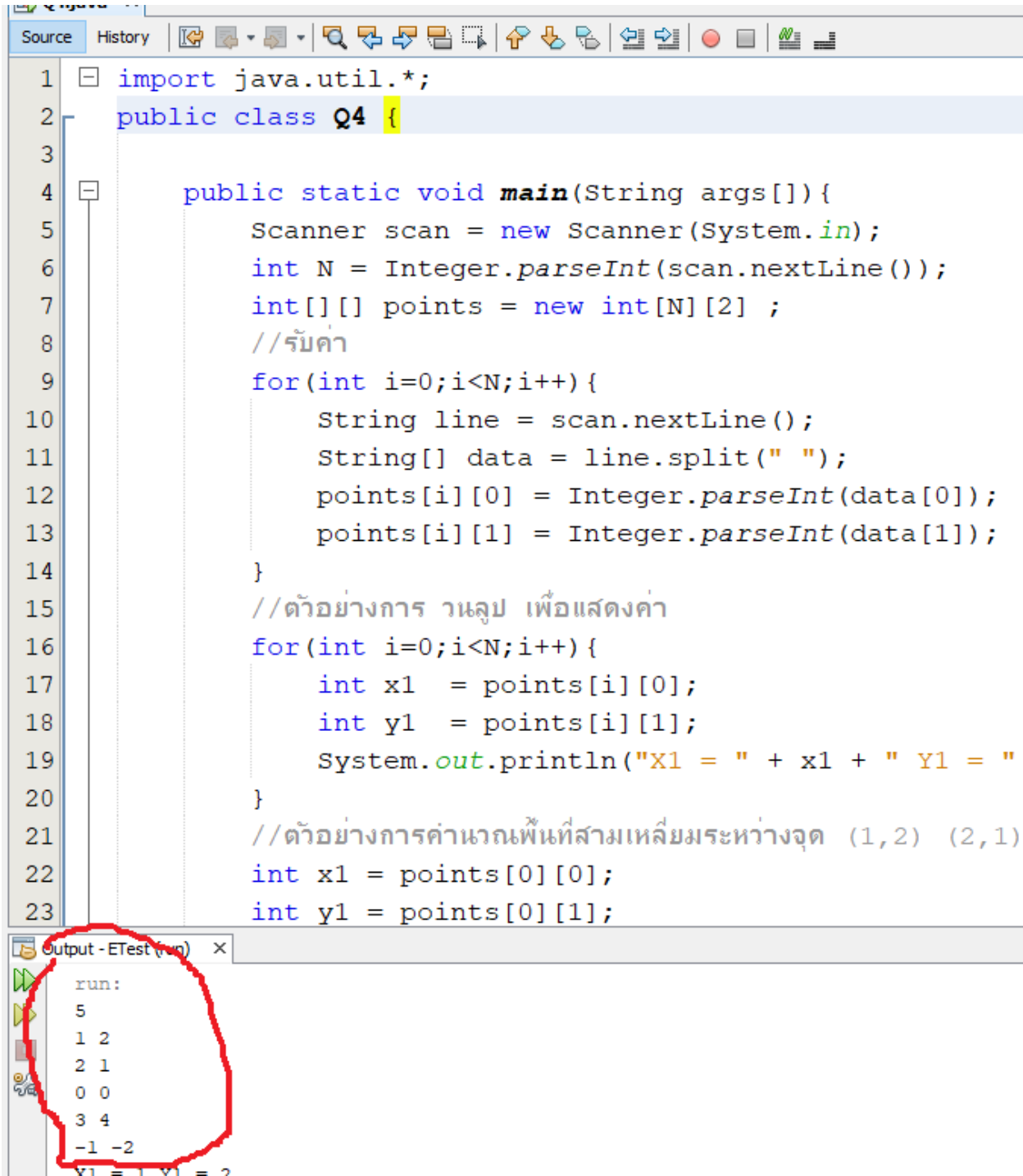
        double area = Math.abs(x1*y2 + x2*y3 + x3*y1 - y1*x2 - y2*x3 - y3*x1)/2.0;

        System.out.println("Area (1,2) (2,1) (0,0)= " +area);

    }

}
```

ตัวอย่างการกรอก input ผ่าน console ด้วยตัวอย่างข้อมูลที่ 1



```
1 import java.util.*;
2 public class Q4 {
3
4     public static void main(String args[]) {
5         Scanner scan = new Scanner(System.in);
6         int N = Integer.parseInt(scan.nextLine());
7         int[][] points = new int[N][2];
8         //รับค่า
9         for(int i=0;i<N;i++){
10             String line = scan.nextLine();
11             String[] data = line.split(" ");
12             points[i][0] = Integer.parseInt(data[0]);
13             points[i][1] = Integer.parseInt(data[1]);
14         }
15         //ตัวอย่างการ วาดรูป เพื่อแสดงค่า
16         for(int i=0;i<N;i++){
17             int x1 = points[i][0];
18             int y1 = points[i][1];
19             System.out.println("X1 = " + x1 + " Y1 = "
20
21         //ตัวอย่างการคำนวณพื้นที่สามเหลี่ยมระหว่างจุด (1,2) (2,1)
22         int x1 = points[0][0];
23         int y1 = points[0][1];
```

Output - ETest (run) x

run:
5
1 2
2 1
0 0
3 4
-1 -2
X1 = 1 Y1 = 2