

โจทย์เตรียมการสิ่งที่ควรรู้สำหรับการสอบ โจทย์นี้คือข้อสอบเมื่อปีที่แล้ว

ให้สร้างโปรเจกต์ PreMid\_<รหัสனிสิต> เอาไว้ทำงานนี้โดยเฉพาะ ให้ project อยู่ใน drive z ให้ขึ้นต้นไฟล์ด้วย ข้อมูลเกี่ยวกับตัวเองดังเช่นแล็บก่อน การส่งให้ส่งเป็น jar ที่มีชอร์สโค้ดอยู่ข้างใน ตั้งชื่อไฟล์เป็น PreMid\_<รหัสனிสิต>.jar ส่วน subject ของอีเมลก็เป็น PreMid\_<รหัสனிสิต>

กำหนดส่งคือ ภายในเวลาเที่ยงคืนของคืนวันศุกร์

โจทย์ข้อนี้ ตอนที่ใช้สอบจริงเมื่อปีที่แล้วนั้น เวลาที่ให้คือสองชั่วโมงครึ่ง ดู sheet และเว็บได้ แต่ห้ามสื่อสาร หรือถามกัน ลองใช้เกณฑ์นี้ฝึกทำดู เราอาจจะเสียเวลาในการเรียนรู้ JUnit เพราะว่ายังไม่ได้สอน(แต่สอนเมื่อปีที่แล้วไปนานก่อนสอบ) แต่ว่ามันทำให้การทดสอบโปรแกรมจัดการง่ายขึ้นนะ

(ทั้งหมด 57 คะแนน)

นักเรียนได้รับมอบหมายให้อู๋ที่เขียนโปรแกรมให้ทำโค้ดต้นแบบของเกมจากหนัง Transformers โดยมีคลาสต่างๆที่ต้องเขียน ดังนี้

คลาสแรกชื่อ Cybertron จะเป็นคลาสที่เป็น template พื้นฐานของหุ่นทุกตัวในเรื่อง คลาสนี้จะเอามาสร้างออบเจกต์ไม่ได้ นั่นคือต้องเป็น abstract class (จะให้ทำเมธอดและตัวแปรที่จำเป็นกับการสอบนี้เท่านั้น ตัวแปรและเมธอดอื่นๆที่น่าจะมีในเกม ไม่ต้องไปคิดถึง) โดยมีตัวแปร (ตัวแปรของทุกคลาสต้องเป็น private ทุกตัว แล้วเราเขียนเมธอดที่ใช้ get กับ set ค่าเอา) ดังนี้

- life เป็นจำนวนเต็มที่แสดงพลังชีวิตของหุ่น
- position เป็นตัวแปรที่เก็บคู่อันดับ x, y ไว้ ซึ่งแสดงถึงตำแหน่งในหน้าจอเกมของตัวหุ่น ซึ่งนักเรียนต้องนิยามคลาส Point ที่จะใช้สร้างออบเจกต์ที่เก็บข้อมูลนี้ขึ้นมาเอง ตำแหน่งนั้นเป็นเลขจำนวนเต็ม
- isVehicle เป็นตัวแปรชนิด boolean ที่มีค่าเป็น true ถ้าตัวหุ่นแปลงร่างเป็นพาหนะอยู่ และเป็น false ถ้าหุ่นยังเป็นรูปร่างหุ่น

เมธอดต่างๆสำหรับคลาส Cybertron เป็นเมธอดที่ไม่สามารถระบุโค้ดได้ ดังนี้

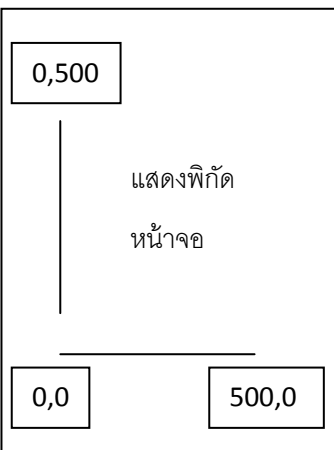
- public void moveHorizontal(int x); บังคับการเคลื่อนที่ไปด้านข้าง (แกน x) ว่าจะเคลื่อนกี่ช่อง (มีทิศทางบวกลบ ขึ้นกับค่า x) การเคลื่อนที่นั้น ให้จำนวนช่องในการเคลื่อนที่เป็นผลคูณของ x กับเลขอื่น (เลขอะไรนั้นขึ้นอยู่กับคลาสที่ extends Cybertron) ซึ่งค่า x ที่เป็นบวกจะบอกทิศทางว่า + เพิ่มตามแกน X แต่ค่าลบจะเป็นการลดตำแหน่งทางแกน X ไป
- public void moveVertical(int y); บังคับการเคลื่อนที่ไปด้านบนหรือล่าง ว่าจะเคลื่อนกี่ช่อง ทำงานเหมือนกับ moveHorizontal แต่เปลี่ยนเป็นแกน Y แทน

ส่วนเมธอดที่มีโค้ดได้นั้น มีดังนี้

- `public void transform();` เปลี่ยนสถานะของค่า `isVehicle` เท่านั้น
- `getLife(), getX(), getY(), getIsVehicle()` ใช้อ่านค่าตัวแปรที่เราต้องการอ่าน
- `setLife(...), setX(...), setY(...), setIsVehicle(...)` ใช้เขียนค่าตัวแปร

1. จงเขียนโค้ดของคลาส `Cybertron` เท่าที่จะเขียนได้ (12 คะแนน)

ต่อไปเราต้องสร้างคลาสที่สามารถสร้างเป็นออบเจกต์หุ่นยนต์ได้ ให้ `Autobot` เป็นสับคลาสของ `Cybertron` และให้มีตัวแปรเพิ่มมาดังต่อไปนี้



- `spirit` เป็นตัวแปรที่แสดงพลังทางคุณธรรมของหุ่น เป็นจำนวนเต็ม ที่มีค่า 0 ถึง 10  
เมธอดที่ต้องเขียนจะมีดังนี้
- `public void moveHorizontal(int x)` ค่า `x` เมธอดนี้จะคำนวณตำแหน่งในแกน X ของหุ่น โดยตำแหน่งใหม่จะมีค่าเท่ากับ ตำแหน่งในแกน X เดิม บวกหรือลบ (ขึ้นกับทิศทางที่ได้จากค่า `x`) กับค่า `x` หุ่นจะเคลื่อนที่ได้ไม่เกินขอบจอ (ตำแหน่ง 0-500) เท่านั้น และค่าของ `x` จะต้องไม่เกินค่า `spirit`
- `public void moveVertical(int y)` ค่า `y` เมธอดนี้จะคำนวณเหมือน `moveHorizontal` เพียงแต่เปลี่ยนตัวแปรมาใช้ค่าตำแหน่งในแกน Y หุ่นจะเคลื่อนที่ได้ไม่เกินขอบจอ (ตำแหน่ง 0-500) เท่านั้น และค่าของ `y` จะต้องไม่เกินค่า `spirit` ด้วย
- `public void transform()` ให้เรียกเมธอดของซูปเปอร์คลาสใช้ แต่ต้องต่อด้วยการพิมพ์ "Autobot transform." ออกหน้าจอ
- คอนสตรัคเตอร์ ที่รับ parameter ดังนี้
  - ตัวแปร `life` สำหรับเกม `Transformers` นั้น ค่าจะมีได้แค่ 0-20 เท่านั้น ถ้าค่าที่ให้มาไม่ใช่ให้ตั้งเป็น 20 ไว้ก่อน
  - ตำแหน่ง `x` และ `y` สำหรับสร้าง `position` สำหรับเกม `Transformers` นั้น ตำแหน่ง `x` และ `y` จะมีค่าได้ตั้งแต่ 0 ถึง 500 เท่านั้น ถ้าตำแหน่งไหนค่าไม่อยู่ในขอบเขตนี้ ให้ตั้งค่านั้นเป็น 250
  - ตัวแปร `isVehicle` ตัวนี้ทำงานตามปกติ ไม่มีข้อบังคับอะไร
  - ตัวแปร `spirit` มีค่า 0-10 เท่านั้น ถ้าไม่ได้ตามขอบเขตนี้ ให้ตั้งค่านั้นเป็น 5

2. จงเขียนคลาส **Autobot** ให้เรียบร้อยแล้วเขียน **junit** เพื่อทดสอบการทำงานของคอนสตรัคเตอร์ เมธอด **moveHorizontal** และเมธอด **transform** ของคลาสนี้ พยายามให้มีกรณีทดสอบครอบคลุมมากที่สุดเท่าที่จำเป็นและให้ **junit** รันได้ถูกต้อง (27 คะแนน)

ตอนนี้เรามีคลาสที่จะใช้ทำหุ่นยนต์ในเกม **Transformers** เรียบร้อยแล้ว แต่ปรากฏว่า งานที่ได้รับมอบหมายนี้ เกิดต้องเปลี่ยนแปลงกระทันหัน เพราะ เราได้รับลิขสิทธิ์หุ่นจากการ์ตูนญี่ปุ่นมา พร้อมทั้งได้โค้ดจาก **Super Robot Wars** ซึ่งเป็นเกมของทางญี่ปุ่นมาด้วย โดยในตอนนี้เราจะต้องหาทางเอาคลาสต่างๆของ **Transformers** ไปใช้กับ เกม **Super Robot** ให้ได้

เกม **Super Robot** นั้น มีการนิยาม **interface** สำหรับใช้ทำหุ่น ดังนี้ (ก๊อป **interface** นี้ใส่ใน **project** เราซะ)

```
public interface Robot{

    public Point jumpTo(int x, int y); // กระโดดสูงไปตกให้ใกล้ตำแหน่ง x ตามแกน X และ y ตามแกน Y ที่สุด

    //เท่าที่จะกระโดดได้ในครั้งนี้

    //วิธีการกระโดดนั้นขึ้นกับหุ่นแต่ละแบบ

    //รีเทิร์นจุดที่หุ่นอยู่หลังจากกระโดดแล้ว

    public void moveHorizontal(int x); // นิยามเหมือนกับที่เรานิยามในคลาส Cybertron

    public void moveVertical(int y); // นิยามเหมือนกับที่เรานิยามในคลาส Cybertron

}
```

ในการจะเอาหุ่นจาก **Transformers** มาใช้งานในเกม จำต้องให้ **Autobot** เป็น **Robot** ด้วย

- สำหรับ **Autobot** การ **jumpTo** ใช้การ **moveHorizontal** หรือ **moveVertical** ใดๆอย่างหนึ่งโดยเทียบตำแหน่งกับจุดหมายเพื่อหาว่ากระโดดไปทางใด ลงที่จุดใด จะอยู่ใกล้จุดเป้าหมายมากที่สุด การกระโดดต้องไม่เกินขอบจอ

3. เขียนโค้ดเพิ่มเติมใน **Autobot** ให้สามารถทำตัวเป็น **Robot** ได้เรียบร้อย (4 คะแนน)

เอาละมาทำต่อกันอีก มีคลาสหุ่นจากเกม **Super Robot** ให้หนึ่งแบบ

```

class SuperWarpRobo implements Robot{
    private int life;
    private int x,y;
    public getX(){ //ให้นักเรียนเขียนเมธอด get ของทุกตัวประกอบ
    }

    public SuperWarpRobo(int life, int x, int y) {
        super();
        this.life = life;
        this.x = x;
        this.y = y;
    }
    public Point jumpTo(int x, int y){
        this.x = x;
        this.y = y;
        return new Point(x,y);
    }
    public void moveHorizontal(int x){
        if (x<0)
            this.x = this.x -1;
        else if (x == 0)
            //do nothing
        else
            this.x = this.x +1;
        if (this.x > 500)
            this.x = 500;
        if(this.x < 0)
            this.x =0;
    }
    public void moveVertical(int y){
        //ไม่เคลื่อนที่ในแกน Y
    }
}

```

นักเรียนได้รับมอบหมายให้เขียนคลาส CargoShip ซึ่งมีพารามิเตอร์ดังนี้

- อาร์เรย์ขนาด 5 ช่องที่ใช้เก็บ Robot
- position ซึ่งเป็น Point ที่เก็บตำแหน่งของ CargoShip

คลาสนี้มีเมธอด

- public Robot launch(int add\_x, int add\_y) เป็นการเอาหุ่นออกมาจากอาร์เรย์ที่ตำแหน่ง index แรก แล้วให้หุ่นตัวนั้น jumpTo ไปที่ตำแหน่ง add\_x จุดบวกไปทางแกน X และ add\_y จุดบวกไปทางแกน Y จากตำแหน่งที่ CargoShip อยู่ รีเทิร์นพอยเตอร์ที่ชี้ไปที่หุ่นนั้น หุ่นตัวอื่นต้องเลื่อนตำแหน่งเพื่อเข้าแทนที่ตำแหน่ง index ด้วย (เหมือนหุ่นอยู่บนสายพาน เลื่อนเพื่อเตรียม launch ต่อได้)
- คอนสตรัคเตอร์ที่รับพารามิเตอร์ x และ y ให้ position นั้นทั้งตำแหน่ง x และ y มีค่าได้ตั้งแต่ 0 ถึง 500 เท่านั้น ถ้าตำแหน่งไหนค่าไม่อยู่ในขอบเขตนี้ ให้ตั้งค่านั้นเป็น 250 ส่วนอาร์เรย์นั้นให้เป็น อาร์เรย์เปล่าขนาด 5 ช่อง
- public void store(Robot r) เป็นการเอาหุ่นเข้าอาร์เรย์ ที่ตำแหน่งที่ไม่มีหุ่นอยู่เป็นตำแหน่งแรก

4. จงเขียนคลาส CargoShip ขึ้นมาให้มีพารามิเตอร์และเมธอดให้ครบตามที่บอกด้านบนนี้ จากนั้นเขียนเมธอด main ของคลาส CargoShip ขึ้นมา แล้วทำการทดลองสถานการณ์การรันดังนี้ (14 คะแนน)
- สร้าง object ชื่อ whiteBase ที่เป็น CargoShip ขึ้นมา ให้อยู่ที่ตำแหน่ง 0,0
  - สร้าง Autobot และ SuperWarpRobo อย่างละตัว แล้วเอาใส่ไปในอาร์เรย์ของ whiteBase โดยให้ตำแหน่งเป็นตำแหน่งเดียวกับ whiteBase
    - ให้แต่ละตัวมีพลังชีวิต 20
    - สำหรับพวกที่เป็น Cybertron ให้มีค่า spirit 5 และค่า isVehicle เป็น false
    - จากนั้นจับใส่อาร์เรย์ตามลำดับ ให้ Autobot อยู่ตำแหน่ง index แรกและ SuperWarpRobo อยู่ตำแหน่งที่สอง
  - ทำการเรียก launch(10,15) หนึ่งครั้ง เก็บค่ารีเทิร์นจากการ launch ลงในตัวแปรประเภท Robot แล้วพิมพ์ตำแหน่งของ Robot ตัวที่ launch ออกมา
  - เขียนโค้ดให้ตัวออบเจกต์หุ่นที่ได้นั้นเรียกเมธอด transform ให้คอมไพล์ผ่าน

สำหรับตอนสอบนั้น เมื่อหมดเวลา ให้ export ตัวโปรเจกต์และโค้ดทั้งหมดเป็น.jar เอาไว้ที่หน้า desktop เปิดเครื่องและ minimize eclipse ไว้ แล้วออกจากห้องเลย