

ชื่อ-นามสกุล _____ รหัสประจำตัว _____

ปัญหา แสวงจุดร่วม สงวนจุดต่าง [ClassJoin] (8 คะแนน)

[ทรัพยากรการคำนวณ: 1 วินาที, 32 MB]

หลังจากพัฒนาคลาสขึ้นส่วนคอมพิวเตอร์มาสองชิ้นคือหน่วยประมวลผลกลางและเมนบอร์ด เราเริ่มสังเกตเห็นว่ามันมีอะไรที่คล้าย ๆ กัน และเราควรจะใช้ของบางอย่างร่วมกัน มากกว่าจะทำให้ต่างกัน โดยเฉพาะเมธอดสำหรับตรวจสอบความถูกต้องของคุณสมบัติขึ้นส่วน ด้วยเหตุนี้ มันจึงเป็นการดีที่เราจะสร้างคลาสที่มีคุณสมบัติร่วมระหว่างชิ้นส่วนต่าง ๆ และให้คลาสขึ้นส่วนสืบทอดคุณสมบัติไปจากคลาสนี้

ในปัญหานี้ เราจะสร้างคลาส **Component** ขึ้นมา พร้อมกับเขียนคลาส **CPU2** และ **MainBoard2** ใหม่ให้มันสืบทอดคุณสมบัติที่เหมือนกันมาจาก **Component** ซึ่งคลาส **Component** ของเราจะมีคุณสมบัติดังนี้

- มีตัวแปรระดับสมาชิกคลาสเป็น (a) ชื่อรุ่นของชิ้นส่วน เป็น **String**, (b) ประเภทช่องเสียบ เป็นจำนวนเต็ม, (c) ประเภทหน่วยความจำเป็นจำนวนเต็ม และ (d) ตัวแปร **valid** ประเภทบูลีน ซึ่งบันทึกความถูกต้องของคุณสมบัติอุปกรณ์
- การตรวจสอบความถูกต้องของคุณสมบัติ (a) - (c) ทำในลักษณะเดียวกันกับข้อที่ผ่านมา ๆ มา คือทำในเมธอด **isValidSpec** เพื่อคืนค่าไปเก็บไว้ในตัวแปร **valid**
- ตัวสร้างจะรับพารามิเตอร์มาเป็น ชื่อรุ่น ประเภทช่องเสียบ และประเภทหน่วยความจำ เพื่อบันทึกค่าไว้ในตัวแปรระดับคลาสที่สอดคล้องกัน แต่จะ**ไม่**เรียกใช้ **isValidSpec** เพราะจะให้ตัวสร้างของชิ้นส่วนที่สืบทอดคุณสมบัติไปเรียกใช้เมธอดนี้เอง⁵
- มีเมธอด **printInfo()** แต่ลำดับการแสดงผลจะเป็น **valid**, ชื่อรุ่น, ประเภทช่องเสียบ และประเภทหน่วยความจำตามลำดับ หนึ่งค่าต่อบรรทัด

คลาส **MainBoard2** ในข้อนี้จะสืบทอด (**inherit**) คลาส **Component** มา โดยที่มันจะต้องเรียกตัวสร้างของ **Component** มาใช้งาน (ทำได้ด้วยการเรียก **super**) และเรียกใช้ **isValidSpec** แล้วรับผลการตรวจสอบมาเก็บไว้ใน **valid** ทั้งนี้ขอให้เข้าใจด้วยว่า **MainBoard2** จะไม่ทำการเก็บค่าในตัวแปรสมาชิกคลาสใด ๆ เลย คือจะต้องทำผ่านตัวสร้างของ **Component** เท่านั้น

⁵ เหตุที่ทำเช่นนี้ก็เพราะว่าชิ้นส่วนแต่ละแบบอาจจะมีรายละเอียดในการตรวจสอบที่แตกต่างกันไปบ้างเล็กน้อย ซึ่งในที่นี้ **CPU2** คือจุดที่มีความแตกต่าง ในขณะที่ **MainBoard2** จะเรียกใช้ **isValidSpec** ของ **Component** ได้ตรง ๆ

ชื่อ-นามสกุล _____ รหัสประจำตัว _____

คลาส **CPU2** สืบทอดคลาส **Component** มาเช่นเดียวกัน แต่จะมีสิ่งที่แตกต่างไปจาก **Component** ดังนี้

1. มีตัวแปรระดับสมาชิกที่เก็บความเร็วสัญญาณเป็นจำนวนเต็ม
2. มีการเขียนทับ (**Override**) เมธอด **printInfo()** ของคลาสพ่อ โดยใน **printInfo()** ของ **CPU2** จะเรียกใช้ **printInfo()** ของคลาสพ่อ แล้วจึงพิมพ์ความเร็วสัญญาณนาฬิกาตามมา ซึ่งการเรียก **printInfo()** ของคลาสพ่อ ทำได้โดยการเขียนว่า **super.printInfo()**
3. มีการเขียนเมธอด **isValidSpec** ที่รับพารามิเตอร์มาแบบเดียวกันกับคลาส **CPU** ที่เคยทำมาก่อนหน้า อย่างไรก็ตาม มันจะทำการเรียกใช้ **isValidSpec** ของคลาสพ่อ และตรวจเพิ่มเติมเฉพาะความเร็วสัญญาณนาฬิกาเพื่อหาข้อสรุปว่าจะคืนค่า **true** หรือ **false** กลับไป ซึ่งกลยุทธ์ในกรณีเช่นนี้ เราอาจจะเรียก **isValidSpec** ของคลาสพ่อเพื่อตรวจสอบว่าผลลัพธ์ที่ได้เป็น **false** หรือไม่ ถ้าเป็นก็คืนค่า **false** กลับไปได้ทันที แต่ถ้าไม่ใช่ ก็ให้ตรวจสอบค่าความเร็วสัญญาณนาฬิกา ถ้าผิดก็คืน **false** ถ้าถูกต้องก็คือ **true**

สุดท้ายตัวสร้างของ **CPU2** จะมีพารามิเตอร์เหมือนกับ **CPU** และจะเก็บเฉพาะค่าของความเร็วสัญญาณนาฬิกาเพิ่มเติม ส่วนตัวแปรระดับสมาชิกอื่น ๆ ที่สืบทอดมา ให้เป็นหน้าที่ของตัวสร้างของคลาสพ่อ จัดเก็บ เช่นเดียวกับ **MainBoard2** และแน่นอนว่าตัวสร้างของ **CPU2** จะต้องเรียก **isValidSpec** ของตัวเองเพื่อรับค่าผลลัพธ์มาเก็บไว้ใน **valid**

ข้อมูลเข้า

เหมือนข้อที่แล้ว คือเป็นข้อมูลของหน่วยประมวลผลกลางตามด้วยข้อมูลของเมนบอร์ด

ผลลัพธ์

เป็นการเรียกใช้ **printInfo()** ของคลาส **CPU2** และคลาส **MainBoard2** ตามลำดับ

หมายเหตุ เพื่อให้รูปแบบของผลลัพธ์ตรงกับความต้องการของโจทย์ ควรใช้ **println** กับการพิมพ์ข้อมูลความเร็วสัญญาณนาฬิกา

ชื่อ-นามสกุล _____ รหัสประจำตัว _____

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
Ryzen 1800X 3200 2333 4 Gigabyte G82 2333 4	true Ryzen 1800X 2333 4 3200 true Gigabyte G82 2333 4
Ryzen 1800X 3200 -1 4 Zotac PXT -1 4	false Ryzen 1800X -1 4 3200 false Zotac PXT -1 4
Ryzen 1800X 3200 2333 4 MSI Carbon 2333 0	true Ryzen 1800X 2333 4 3200 false MSI Carbon 2333 0
Ryzen 1800X 3200 2333 0 MSI Carbon 2333 3	false Ryzen 1800X 2333 0 3200 true MSI Carbon 2333 3

เกณฑ์การให้คะแนน โปรแกรมจะต้องทำถูกต้องอย่างน้อย 70% ของชุดทดสอบ จึงจะได้คะแนน

ชื่อ-นามสกุล _____ รหัสประจำตัว _____

โครงสร้างโค้ดสำหรับข้อนี้

```
import java.util.Scanner;

class Component {
    // Enter your code here
}

class CPU2 extends Component {
    // Enter your code here
}

class MainBoard2 extends Component {
    // Enter your code here
}

public class ClassJoin {
    // Do not change the main method.
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        String model = scan.nextLine();
        int clocks = scan.nextInt();
        int socket = scan.nextInt();
        int memType = scan.nextInt();

        if(model.length() == 0 && socket % 2 == 0)
            model = null;
        Component cpu = new CPU2(model, clocks, socket, memType);
        cpu.printInfo();

        scan.nextLine(); // skip new line character
        model = scan.nextLine();
        socket = scan.nextInt();
        memType = scan.nextInt();
        if(model.length() == 0 && socket % 2 == 0)
            model = null;
        Component mb = new MainBoard2(model, socket, memType);
        mb.printInfo();
    }
}
```