

1. จงเขียนโปรแกรมรับเลขจำนวนเต็ม 2 จำนวน ซึ่งแทนความกว้างและความยาวของรูปสี่เหลี่ยม จากนั้นให้พิจารณาว่าข้อมูลที่ได้รับมานั้น ใช้รูปสี่เหลี่ยมด้านเท่าหรือไม่
 - ถ้าใช่ให้คำนวณพื้นที่ของรูปสี่เหลี่ยมนั้น พร้อมแสดงผลลัพธ์ทางหน้าจอ
 - ถ้าไม่ใช่แสดงข้อความ “No” แล้วจบการทำงานของโปรแกรม

โดยกำหนดให้มี method ชื่อ isEqual() และ area() ดังนี้

- `public static void isEqual(int w, int h) {...}` // ส่งข้อมูลไปเช็คค่าเท่ากันไหมถ้าไม่เท่าปรี้น No , ถ้าเท่าปรี้น Yes และเรียก method calArea() ทำงาน
- `public static int calArea(int x) {...}` // คำนวณพื้นที่รูปสี่เหลี่ยมแล้วแสดงผลทางหน้าจอ

ตัวอย่าง

input	output
20 20	Yes Area is 400
12 20	No

2. จงเขียนโปรแกรมรับจำนวนเต็ม 3 จำนวน จากนั้นตรวจสอบพร้อมกับแสดงผลลัพธ์ว่า มีสองจำนวนใดๆในข้อมูลที่รับเข้ามา ที่มีค่าเท่ากันหรือไม่
 - ถ้ามีให้แสดงเลข 2 ตัวไหนบ้างที่เหมือนกันทางหน้าจอ
 - ถ้าไม่มีคู่ใดเลยที่เหมือนกันให้แสดงข้อความ “Not Equal”

หมายเหตุ : (ตัวอย่าง method สำหรับโจทย์ข้อนี้)

- `public static void checkEqual(int a, int b, int c){...}` // รับข้อมูลมาเพื่อจับคู่ตัวเลขที่ต้องการทดสอบส่งไปให้ method isEqual() ทำงาน
- `public static boolean isEqual(int x, int y) {...}` // เพื่อทดสอบว่าข้อมูลที่ส่งให้มามีค่าเท่ากันหรือไม่ ถ้าเท่าคืนค่า true ไม่เท่าคืนค่า false

ตัวอย่าง

input	output	หมายเหตุ
1 2 3	Not Equal	ไม่มีตัวไหนเท่ากัน
2 2 3	num(1) = num(2)	เลขตัวที่ 1 เท่ากับตัวที่ 2
3 3 3	num(1) = num(2) num(1) = num(3) num(2) = num(3)	เลขตัวที่ 1 เท่ากับตัวที่ 2 เลขตัวที่ 1 เท่ากับตัวที่ 3 เลขตัวที่ 2 เท่ากับตัวที่ 3

3. จงเขียนโปรแกรมรับจำนวนเต็มที่แทนวัน เดือน ปี (ค.ศ.) เพื่อแสดงผลการคำนวณว่า วันเดือนปีที่ป้อนเข้ามา นั้นเป็นวันที่เท่าไรของปี เช่น วันที่ 1 มกราคม 2007 เป็นวันที่ 1 ของปี , วันที่ 31 ธันวาคม 2007 เป็นวันที่ 365 ของปี ในขณะที่ 31 ธันวาคม 2008 เป็นวันที่ 366 ของปี (เพราะปีนี้เป็นเดือนกุมภาพันธ์มี 29 วัน)

หมายเหตุ 1 : การตรวจสอบว่าปีใดๆเป็นปีที่เดือนกุมภาพันธ์มี 29 วันหรือไม่ ทำดังนี้

- ถ้าเลขปีใด(ค.ศ)หารด้วย 400 ลงตัว ปีนั้นจะเป็นปีที่เดือนกุมภาพันธ์มี 29 วัน หรือ
- ถ้าเลขปีใด(ค.ศ)หารด้วย 4 ลงตัว แต่หารด้วย 100 ไม่ลงตัว ปีนั้นเดือนกุมภาพันธ์ก็มี 29 วัน
- นอกเหนือจาก 2 เงื่อนไขข้างต้น จะเป็นปีที่เดือนกุมภาพันธ์ 28 วัน

หมายเหตุ 2 : ตัวอย่าง method ของโปรแกรมนี้นี้

- `public static intcalNumDay(int d, int m ,int y) {...}` //รับข้อมูลวัน เดือน ปี เพื่อคำนวณว่าเป็นวันที่เท่าไรของปี แล้วคืนผลลัพธ์ที่คำนวณได้กลับไป
- `public static intnumDayofMonth(int m) {...}` //รับหมายเลขเดือนเข้ามาเพื่อคำนวณว่าเดือนนั้นมีกี่วัน และคืนจำนวนวันที่ได้กลับไป
- `public static booleanisLeapYear(int y) {...}` //รับปีค.ศ. มาเพื่อทดสอบว่าปีนั้นเป็นปีที่เดือนกุมภาพันธ์มี 29 วันหรือไม่ ถ้าใช่คืนค่า true ไม่ใช่คืนค่า false

ตัวอย่าง

input	output
1 1 2007	1
31 12 2007	365
31 12 2008	366

4. จงเขียนโปรแกรมเพื่อรับค่าจำนวนเต็ม 3 จำนวน ได้แก่ ชั่วโมง(h), นาที(m)และหมายเลขที่แทนรูปแบบการแสดงผลที่ต้องการของเวลาใดๆ แล้วแสดงผลเวลาดังกล่าวในรูปแบบต่างๆ ดังนี้
- รูปแบบที่ 1 แสดงเวลาในรูปแบบ 24 ชั่วโมง เช่น ถ้า $h = 13$, $n = 0$ จะแสดงเป็น 13:00
 - รูปแบบที่ 2 แสดงเวลาในรูปแบบ 12 ชั่วโมง เช่น ถ้า $h = 13$, $n = 0$ จะแสดงเป็น 1:00PM
 - รูปแบบที่ 3 แสดงเวลาในรูปแบบ 24 ชั่วโมง แต่เป็นคำพูด เช่น ถ้า $h = 13$, $n = 0$ จะแสดงเป็น 13 o'clock and 0 minute(s)

หมายเหตุ : ตัวอย่าง method ของโปรแกรมนี้นี้

- `public static void selectFormat(int h , int m , int f){...}` // รับข้อมูลทั้ง 3 รายการเพื่อเลือก method ที่ตรงกับรูปแบบที่ผู้ใช้เลือกมาทำงาน พร้อมส่งข้อมูล ชั่วโมง และ นาทีให้ method นั้นด้วย
- `public static void normal(int h, int m){...}` // รับค่าชั่วโมง นาที แล้วแสดงผลลัพธ์ตามรูปแบบที่ 1
- `public static void am_pm(int h, int m){...}` // รับค่าชั่วโมง นาที แล้วแสดงผลลัพธ์ตามรูปแบบที่ 2
- `public static void verbose(int h, int m){...}` // รับค่าชั่วโมง นาที แล้วแสดงผลลัพธ์ตามรูปแบบที่ 3

ตัวอย่าง

input	output
13 0 1	13:00
8 0 2	8:00AM
13 1 3	13 o'clock and 1 minute(s)

5. เลขบัตรเครดิตจะมี check digit (เลขไว้ตรวจสอบว่าหมายเลขบัตรถูกจริงไหม) จงเขียนโปรแกรมเพื่อรับหมายเลขบัตรเครดิต 1 หมายเลข (มี 7 หลักเสมอ) แล้วทำการตรวจสอบว่าหมายเลขดังกล่าวถูกต้องหรือไม่

- ถ้าถูกให้แสดงข้อความ “valid”
- ถ้าไม่ถูกให้แสดงข้อความ “invalid”

วิธีการตรวจสอบเลขบัตรเครดิต

1. ทำการถอดรหัสเลขแต่ละตัวโดยพิจารณาจากตำแหน่งของตัวเลขโดดแต่ละตัว ดังนี้

เช่นหมายเลขบัตรเป็น 2315778

หมายเลข	8	7	7	5	1	3	2
ตำแหน่งที่	1	2	3	4	5	6	7

- เลขไหนอยู่ตำแหน่งที่เป็นเลขคู่จะถอดรหัสได้เป็นเลขนั้นเลยเช่น 8 จะได้ 8 , 7 ก็เป็น 7
 - เลขไหนอยู่ตำแหน่งที่เป็นเลขคี่ จะต้องนำเลขนั้นไปคูณ 2 แล้วเช็คค่าผลคูณเกิน 10 ไหม
 - ถ้าไม่เกิน ผลการถอดรหัสก็คือเลขนั้นเช่น ตำแหน่งที่ 6 เป็นเลข 3 ดังนั้น จะถอดรหัสได้เป็น 6 (มาจาก $3 * 2$)
 - ถ้าเกิน ผลการถอดรหัสก็คือเอาเลขแต่ละหลักมาบวกกัน เช่น ตำแหน่งที่ 2 เป็นเลข 7 เมื่อนำไปคูณ 2 ได้ 14 ผลการถอดรหัสคือ 5 (มาจาก $1 + 4$)
2. เมื่อถอดรหัสเลขครบทุกตำแหน่งแล้ว ให้ทำการบวกตัวเลขที่ถอดรหัสแล้วเข้าด้วยกัน (sum)
3. พิจารณาผลบวกที่ได้จากข้อ 2 ว่า หมายเลขบัตรถูกไหม ด้วยเงื่อนไข
- ถ้าผลบวกหารด้วย 10 ลงตัว แสดงว่าเป็นหมายเลขที่ถูกต้อง
 - ถ้าผลบวกหารด้วย 10 ไม่ลงตัว แสดงว่าเป็นหมายเลขบัตรที่มีความผิดพลาด

ตัวอย่างการคำนวณ

เช่นหมายเลขบัตรเป็น 2315778

หมายเลข	8	7	7	5	1	3	2
ตำแหน่งที่	1	2	3	4	5	6	7
ถอดรหัส	8	5	7	1	1	6	2

ทำการหาผลรวมเลขทุกตัวหลังถอดรหัสได้เป็น $= 8 + 5 + 7 + 1 + 1 + 6 + 2 = 30$

ซึ่ง 30 หารด้วย 10 ลงตัวดังนั้น valid

เช่นหมายเลขบัตรเป็น 1234567

หมายเลข	7	6	5	4	3	2	1
ตำแหน่งที่	1	2	3	4	5	6	7
ถอดรหัส	7	3	5	8	3	4	1

ทำการหาผลรวมเลขทุกตัวหลังถอดรหัสได้เป็น $= 7 + 3 + 5 + 8 + 3 + 4 + 1 = 31$

ซึ่ง 31 หารด้วย 10 ไม่ลงตัวดังนั้น invalid

หมายเหตุ : ตัวอย่าง method ของโปรแกรมนี้

- `public static void checkDigit(intnum){...}` // รับตัวเลขใดๆ เพื่อแยกเลขโดดในแต่ละตำแหน่ง ส่งไปถอดรหัสใน method decode แล้วรับค่าที่ได้มาหาผลรวมสะสมไปเรื่อยๆ
- `public static int decode(intnum , intpos) {...}` // รับตัวเลขใดๆ พร้อมหมายเลขหลักของ ตัวเลขนั้นเพื่อคำนวณค่าหลังถอดรหัส แล้วคืนค่ากลับ
- `public static void checkCard(int sum) {...}` //รับค่าผลรวมตัวเลขหลังถอดรหัสมาทำการ ตรวจสอบว่ารหัสบัตรถูกต้องหรือไม่ ถ้าถูกปรี้น valid ถ้าไม่ถูกปรี้น invalid

ตัวอย่าง

input	output
2315778	valid
1234567	invalid