

## ปัญหา ถังเก็บน้ำ [WaterTank]

ในงานนี้ เราจะทำการคำนวณเกี่ยวกับถังเก็บน้ำ โดยจะมีการคำนวณความจุถังและติดตามปริมาณน้ำด้วยวัตถุจากคลาส **WaterTank** ตามข้อกำหนดดังนี้

1. คลาส **WaterTank** มีตัวสร้างที่รับพารามิเตอร์มาสามค่าคือความกว้าง ความยาว และความสูง ตามลำดับซึ่งเป็นจำนวนเต็มทั้งหมด และตัวสร้างนี้จะคำนวณความจุถังเก็บน้ำด้วยสูตร กว้าง  $\times$  ยาว  $\times$  สูง และเก็บความจุถังไว้ในตัวแปรสมาชิกคลาสชื่อ **capacity** และมีตัวแปรสมาชิกคลาสอีกตัวชื่อ **volume** ซึ่งเก็บปริมาณน้ำ ณ ปัจจุบันไว้ โดยในตอนแรกให้มีค่าเป็นศูนย์

2. คลาส **WaterTank** มีเมธอด **fill** โดยเมธอดนี้รับพารามิเตอร์มาหนึ่งตัวเป็นจำนวนเต็ม และเมธอดจะเติมน้ำเข้าไปในถังเก็บด้วยปริมาณตามที่ระบุไว้ในพารามิเตอร์ โดยเมธอดนี้จะทดสอบก่อนว่าถ้าเติมน้ำไปแล้วปริมาณน้ำจะล้นถังเก็บหรือไม่ ซึ่งถ้าล้นเมธอดจะพิมพ์ว่า **“Cannot fill the tank”** แล้วขึ้นบรรทัดใหม่ จากนั้นเมธอด **fill** จะคืนค่า **-1** กลับไป<sup>1</sup> แต่หากเติมได้เพราะไม่ล้น เมธอดจะปรับปริมาณน้ำ (ค่า **volume**) โดยบวกปริมาณน้ำจากพารามิเตอร์เข้าไป และคืน (**return**) ปริมาณน้ำที่ปรับแล้วคืนไปเป็นผลลัพธ์

หมายเหตุ กำหนดให้หน่วยความจุในตัวแปร **volume** และพารามิเตอร์นี้มีหน่วยที่เหมือนกัน ทำให้น้ำค่ามาบวกกันได้โดยตรง

สำหรับเมธอด **main** จะอยู่ในอีกคลาสหนึ่งและจะรับขนาดถังเก็บน้ำมาเพื่อสร้างวัตถุชนิด **WaterTank** จากขนาดที่ได้รับมานั้นและเมธอด **main** จะทำการเรียกเมธอด **fill** ของวัตถุดังกล่าว 5 ครั้งเพื่อเติมน้ำ และพิมพ์ผลลัพธ์ที่คืนกลับมาจากเมธอด **fill** มาเป็นผลลัพธ์

จงเขียนโปรแกรมที่รับขนาดของถังเก็บน้ำและปริมาณน้ำที่จะเติมตามข้อกำหนดที่ระบุไว้ข้างบน

---

<sup>1</sup> กรณีนี้ เราจะถือว่าไม่มีการเติมน้ำเข้าไปในถัง

## รูปแบบข้อมูลเข้า

บรรทัดแรก	เป็นจำนวนเต็มสามค่าคือความกว้าง ความยาว และความสูงของถัง เก็บน้ำตามลำดับ โดยที่เลขทั้งสามเป็นจำนวนเต็มบวกค่าไม่เกิน <b>1,000</b>
บรรทัดที่สอง ถึงหก	เป็นเลขจำนวนเต็มบวกที่แทนปริมาณน้ำที่จะเติมเข้าไปในถัง หนึ่งค่า ต่อบรรทัด (การเติมน้ำเรียงตามบรรทัดจากก่อนไปหลัง)

## รูปแบบผลลัพธ์

เป็นผลลัพธ์จากการเรียกเมธอด **fill** แต่ละครั้ง เรียงจากครั้งแรกไปจนถึงครั้งสุดท้าย  
โดยจะเป็นผลลัพธ์จากค่าที่คืนมาจากเมธอด **fill** ซึ่งหากเติมน้ำเข้าไปไม่ได้จะมีการพิมพ์  
ข้อความโดยเมธอด **fill** ก่อนจะพิมพ์ค่าที่คืนกลับมา

## ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
2 5 3 4 20 10 5 1	4 24 Cannot fill the tank -1 29 30
2 5 8 100 50 20 5 10	Cannot fill the tank -1 50 70 75 Cannot fill the tank -1

## เกณฑ์การให้คะแนน

โปรแกรมจะต้องทำงานถูกต้องอย่างน้อย 2 กรณีในชุดทดสอบ จึงจะได้คะแนน  
[มีโครงสร้างโค้ดให้ในหน้าถัดไป]

## โครงสร้างโค้ดคำตอบ

```
import java.util.Scanner;

class WaterTank {
    // Enter your code here.
}

public class TankObserver {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        int width = scan.nextInt();
        int length = scan.nextInt();
        int height = scan.nextInt();
        WaterTank tank = ???;

        for(int i = 0; i < 5; ++i) {
            int amount = scan.nextInt();
            //System.out.println(???);
        }
    }
}
```