

ปัญหา เส้นทางรถบัส 1 [BusRoute1]

(หมายเหตุ เป็นไปได้ว่าเราอาจจะทำข้อต่อไปได้โดยไม่ต้องทำข้อนี้ให้เสร็จอย่างสมบูรณ์ เป็นต้นว่าต่อให้คุณทำเมธอด `getStopInfo` ของข้อนี้ไม่สำเร็จและข้อนี้ได้ศูนย์คะแนน ก็ไม่ได้หมายความว่าเราจะทำข้อต่อไปให้ได้คะแนนเต็มไม่ได้)

เส้นทางรถโดยสารนอกจากจะมีสถานีต้นทางและปลายทาง (หรือบางที่เรียกว่าป้ายหรือศาลาก็มี) ก็มีสถานีระหว่างทางที่รถแวะจอดเพื่อรับส่งผู้โดยสารด้วย ซึ่งสถานีแต่ละแห่งก็จะมีชื่อและมีระยะบอกจากสถานีก่อนหน้านี้ด้วย ซึ่งชื่อนั้นเอาไว้ใช้สำหรับการสื่อสาร ส่วนระยะทางเอาไว้ใช้ในการคำนวณว่าผู้โดยสารใช้บริการเป็นระยะทางเท่าใด เพื่อที่จะสามารถคิดค่าโดยสารได้ตามกรอบที่กฎหมายกำหนด

ในโปรแกรมนี้ เราต้องการทราบว่าเมื่อระบุลำดับของสถานีที่ผู้โดยสารขึ้นลงไปให้ ชื่อสถานีคืออะไรและระยะทางทั้งหมดที่ผู้โดยสารเดินทางคือเท่าใด

เพื่อให้การจัดการข้อมูลเป็นไปอย่างเป็นระบบและสามารถจัดเป็นบริการด้านข้อมูลจากวัตถุหนึ่งไปให้อีกวัตถุหนึ่งได้ เราจะสร้างคลาส `Route` ซึ่งมีข้อกำหนด¹ ของคลาสเป็นดังนี้

1. ตัวสร้าง (constructor) ของคลาสจะรับอาร์เรย์มาสองตัวซึ่งมีความยาวเท่ากัน ตัวแรกคือชื่อของสถานีต่าง ๆ เรียงจากสถานีแรกไปจนถึงสถานีสุดท้าย ส่วนอาร์เรย์ตัวที่สองคือระยะทางจากสถานีก่อนหน้านี้ เช่นถ้าพูดถึงค่าที่เป็นระยะทางของสถานีที่ 7 จะหมายถึงระยะทางจากสถานีที่ 6 มาถึงสถานีที่ 7 ทั้งนี้กำหนดให้ค่าระยะทางของสถานีแรกคือศูนย์ เพราะไม่มีสถานีก่อนหน้านี้ กำหนดเพิ่มเติมว่าชื่อสถานีต่าง ๆ เป็นสตริงที่ไม่มีช่องว่าง และค่าระยะทางเป็นจำนวนเต็ม มีหน่วยเป็นกิโลเมตร ซึ่งตัวสร้างจะต้องรับข้อมูลเหล่านี้ไปเก็บไว้ในตัวแปรสมาชิกคลาสสำหรับใช้งานต่อไปในภายหลัง
2. มีเมธอด `getStopInfo` ซึ่งจะรับพารามิเตอร์มาเป็นหมายเลขลำดับของสถานีเป็นจำนวนเต็ม โดยสถานีแรกจะมีหมายเลขลำดับเป็นเลขหนึ่ง หากหมายเลขลำดับที่ได้มา

¹ คำว่า “ข้อกำหนด” ก็คือ “สิ่งที่กำหนดให้ทำตาม ถ้าไม่ทำตามถือว่าผิด” ไม่ใช่เข้าไปเขียนอะไรอย่างอื่นมาแล้วได้คำตอบถูกแต่ผิดข้อกำหนด พอโดนหักคะแนนก็มาโวยวายว่า “อาจารย์ไม่ได้ห้าม จะมาหักคะแนนกันได้ไง” ดังนั้นขอให้เข้าใจคำว่าข้อกำหนดตามนี้ด้วย

ทางพารามิเตอร์อยู่ในขอบเขตของหมายเลขสถานีที่อยู่ในเส้นทางนี้ เมธอดจะพิมพ์ชื่อของสถานีออกมาพร้อมกับระยะทางจากสถานีก่อนหน้านี้ แต่หากหมายเลขลำดับที่ได้มานอกขอบเขต เมธอดนี้จะพิมพ์ว่า **invalid number** ออกมาแทน (กล่าวคือ เมธอดนี้จะต้องทำการตรวจสอบความถูกต้องด้วยตัวเอง อย่าไปทำในเมธอด **main**)

สำหรับตัวเมธอด **main** จะแยกอยู่นอกคลาส **Route** ไปอยู่ในคลาสหลักของโจทย์ ซึ่งจะรับข้อมูลสถานีของเส้นทางทั้งหมด และรายการหมายเลขลำดับสถานีที่ต้องการทราบข้อมูล ซึ่งในที่นี้กำหนดให้พิมพ์ข้อมูลสถานีที่ต้องการออกมาด้วยเมธอด **getStopInfo**

จงเขียนโปรแกรมเส้นทางรถบัสที่รับข้อมูลต่าง ๆ มาเพื่อสร้างวัตถุ **Route** และให้ **main** เรียกใช้บริการจากเมธอด **getStopInfo** ของวัตถุชนิด **Route** เพื่อให้เมธอดดังกล่าวพิมพ์ข้อมูลสถานีที่สนใจออกมา

รูปแบบข้อมูลเข้า

บรรทัดแรก	เป็นค่า N ซึ่งแทนจำนวนสถานีบนเส้นทางรถบัส โดยที่ $N \leq 500$
อีก N บรรทัดถัดมา	เป็นข้อมูลของแต่ละสถานีหนึ่งบรรทัดต่อสถานี ซึ่งในแต่ละบรรทัดจะเป็นชื่อสถานีและระยะทางจากสถานีก่อนหน้านี้
บรรทัดที่ N+1	เป็นค่า K ซึ่งแทนจำนวนหมายเลขลำดับสถานีที่ต้องการส่งไปสอบถามข้อมูลจากวัตถุ Route
บรรทัดถัดมา	เป็นหมายเลขลำดับสถานีที่ต้องการสอบถามข้อมูล ซึ่งหมายเลขลำดับนี้อาจจะมีค่าอยู่นอกขอบเขตหมายเลขสถานีที่มีอยู่จริงในเส้นทางรถบัส

รูปแบบผลลัพธ์

มี K บรรทัด ซึ่งเป็นสิ่งที่เมธอด **getStopInfo** พิมพ์ออกมาตามหมายเลขลำดับสถานีที่มีการสอบถามเข้ามาทางข้อมูลเข้า

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
7 ThaPhra 0 BangPhai 2 BangWa 5 Phetkasem 3 Phasicharoen 6 BangKhae 4 LakSong 1 8 4 3 2 0 -1 8 7 1	Phetkasem 3 BangWa 5 BangPhai 2 invalid number invalid number invalid number LakSong 1 ThaPhra 0
9 ThaPhra 0 Itsaraphap 3 Sanamchai 5 SamYot 2 WatMangkon 6 HuaLamphong 2 SamYan 3 SiLom 5 Lumphini 5 12 -10 10 1 0 9 8 7 2 4 3 5 6	invalid number invalid number ThaPhra 0 invalid number Lumphini 5 SiLom 5 SamYan 3 Itsaraphap 3 SamYot 2 Sanamchai 5 WatMangkon 6 HuaLamphong 2

เกณฑ์การให้คะแนน

โปรแกรมจะต้องทำงานถูกต้องอย่างน้อย 30% ของชุดทดสอบ จึงจะได้คะแนน

[มีโครงสร้างโค้ดให้ในหน้าถัดไป]

โครงสร้างโค้ดสำหรับการทำข้อสอบ แนะนำให้เติมเต็มคลาส Route ตามข้อกำหนด
ส่วนในเมธอด main ให้เน้นแก้โค้ดตรงที่เป็น ???

```
import java.util.Scanner;

class Route {
    // Enter your code here.
}

public class BusRoute1 {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        final int N = scan.nextInt();
        String[] names = ???;
        int[] distances = ???;
        for(???) {
            names[???] = scan.next();
            distances[???] = scan.nextInt();
        }

        Route ??? = ???; // Create a Route object here.
        final int K = scan.nextInt();
        for(int i = 0; i < K; ++i) {
            int ??? = scan.nextInt();
            ???;
        }
    }
}
```