

## ชนแร่ด้วยรถไฟ

1 second, 256 MB

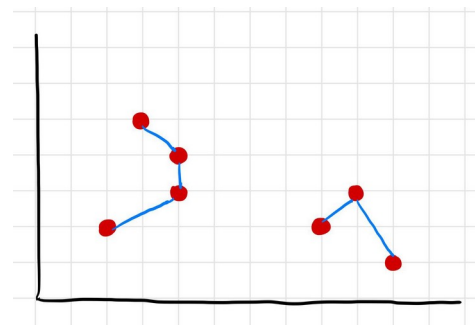
ที่อาณาจักรแห่งหนึ่ง มีเมืองแร่จำนวน  $N$  เมือง เมืองที่  $i$  สำหรับ  $1 \leq i \leq N$  มีตำแหน่งอยู่ที่พิกัด  $(X_i, Y_i)$  ระยะทางระหว่างเมืองแร่จะคิดตามระยะทางแบบยูคลิด (Euclidean distance) นั่นคือ เมืองที่  $i$  และ  $j$  อยู่ห่างกัน  $\sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2}$  เราต้องการเชื่อมเมืองเข้าด้วยกันโดยสร้างทางรถไฟที่เป็นเส้นตรงหลาย ๆ สาย ทางรถไฟแต่ละสายจะเริ่มที่เมืองหนึ่งและสิ้นสุดที่อีกเมือง ในการขนของด้วยรถไฟระหว่างเมืองนั้นอาจจะทำได้โดยการเดินทางผ่านทางรถไฟหลายสายต่อเนื่องกันได้

เป้าหมายของการเชื่อมเมืองเข้าด้วยกันเพื่อที่จะสร้างศูนย์กำจัดการพิษและใช้รถไฟขนสารพิษไปบำบัดที่ศูนย์เหล่านี้ อาณาจักรดังกล่าวมีงบประมาณสร้างศูนย์กำจัดการพิษสองที่ซึ่งจะสร้างที่เมืองสองเมือง คุณสังเกตว่าถ้าสามารถสร้างศูนย์กำจัดการพิษได้สองศูนย์ก็ไม่จำเป็นต้องเชื่อมทุกเมืองเข้าด้วยกันก็ได้ แค่เชื่อมให้ทุก ๆ เมืองสามารถเดินทางผ่านทางรถไฟไปยังสองเมืองที่จะเลือกสร้างศูนย์ได้ก็เพียงพอ

เนื่องจากค่าใช้จ่ายในการสร้างทางรถไฟแปรผันตรงกับระยะทาง เราจะคิดค่าใช้จ่ายเท่ากับระยะทางระหว่างเมือง ให้คุณเขียนโปรแกรมเพื่อหาค่าใช้จ่ายน้อยที่สุดที่เชื่อมเมืองเข้าด้วยกันโดยรับประกันว่ามีวิธีที่จะเลือกเมืองสองเมืองเพื่อสร้างศูนย์กำจัดการพิษและทุกเมืองมีเส้นทางทางรถไฟที่จะขนสารพิษไปยังศูนย์กำจัดการพิษได้อย่างน้อยหนึ่งศูนย์

พิจารณาตัวอย่างเมืองจำนวน 7 เมืองที่มีพิกัดต่อไปนี้

i	$X_i$	$Y_i$
1	2	2
2	8	2
3	9	3
4	4	3
5	10	1
6	3	5
7	4	4



รูปด้านซ้ายแสดงวิธีการเชื่อมต่อเมืองเข้าด้วยกันด้วยรางรถไฟจำนวน 5 เส้น โดยสังเกตว่าถ้าเลือกให้เมืองที่ 1 และเมืองที่ 5 เป็นศูนย์กำจัดการพิษ ทุก ๆ เมืองจะสามารถขนสารพิษไปยังเมืองทั้งสองผ่านทางรางรถไฟที่สร้างได้ ระยะทางรวมคือ 8.30056

### ข้อมูลนำเข้า

บรรทัดแรกระบุจำนวนเต็ม  $N$  ( $1 \leq N \leq 500$ ) จากนั้นอีก  $N$  บรรทัดระบุพิกัดของเมืองแร่แต่ละเมืองเป็นคู่ของจำนวนเต็ม มีค่าระหว่าง 0 ถึง 1,000

### ข้อมูลส่งออก

มีหนึ่งบรรทัด เป็นจำนวนจริงแทนค่าใช้จ่าย ค่าตอบที่ถูกต้องจะต้องมีความผิดพลาดไม่เกิน 0.001 (ไม่ต้องกังวลตอนปัดค่า โดยมากจะไม่มีปัญหา อย่าลืมอ่านรายละเอียดเกี่ยวกับการเขียนโปรแกรมท้ายโจทย์)

### ตัวอย่าง 1

<u>Input</u>	<u>Output</u>
7 2 2 8 2 9 3 4 3 10 1 3 5 4 4	8.3006

หมายเหตุคำตอบที่มีความละเอียดสูงขึ้นคือ 8.30056308 แต่คำตอบใด ๆ ที่แตกต่างกันไม่เกิน 0.001 จะถือว่าเป็นคำตอบที่ถูกต้องทั้งหมด

### ตัวอย่าง 2

<u>Input</u>	<u>Output</u>
5 10 0 20 0 35 0 45 0 55 0	30.0000

### หมายเหตุการโปรแกรม

ในการเก็บระยะทาง ให้ใช้ตัวแปรประเภท double สามารถใช้ฟังก์ชัน sqrt ในการหารากที่ 2 ได้ ในการใช้ ถ้าเขียนแบบ C++ ให้

```
#include <cmath>
```

ถ้าอยากจะเขียนแบบ C ใช้

```
#include <math.h>
```

ในการพิมพ์ ถ้าใช้ printf ในการพิมพ์ double ให้ระบุจำนวนทศนิยมให้มากกว่า 3 ตำแหน่ง เช่นสั่ง  
printf("%.5f", sol);

ถ้าใช้ cout ในการพิมพ์ ให้สั่ง

```
cout.precision(10);
```

ก่อนพิมพ์ เพื่อให้ผลลัพธ์ออกมาประมาณ 5 ตำแหน่งเป็นต้น