



การแข่งขันล่าสมบัติในแกรนด์ไลน์ (Grand Line Treasure Race)

(8 mb, 1 วินาที)

ลูฟี่และลูกเรือหมวกฟางตัดสินใจแยกกันเดินทางด้วยเรือ 2 ลำเพื่อค้นหาสมบัติในแกรนด์ไลน์ โดยมีการแข่งขันกันว่าใครจะได้สมบัติมูลค่ามากกว่ากันเมื่อเรือทั้งสองลำเดินทางด้วยความเร็วเท่ากัน คือ - **หนึ่งช่องต่อหนึ่งหน่วยเวลา**

หน้าที่ของคุณคือ เขียนโปรแกรมเพื่อคำนวณหาพิกัดสุดท้าย และมูลค่าของสมบัติที่เก็บได้ของเรือแต่ละลำ

ระบบพิกัด - แผนที่แกรนด์ไลน์

แผนที่เป็นตารางสองมิติ มีพิกัดเป็นจำนวนเต็มที่ไม่เป็นลบ

- มุมซ้ายบน คือ (0, 0) โดยที่ (row, column) = (0, 0)
- row เพิ่มขึ้นลงมาด้านล่าง (เหนือ → ใต้)
- column เพิ่มขึ้นไปทางขวา (ตะวันตก → ตะวันออก)

การเดินทางเรือและคำสั่ง

แต่ละเรือมี:

- จุดเริ่มต้น (row, column)
- ชุดคำสั่งการเดินทางเรือ ประกอบด้วย:
 - ทิศทาง: N (เหนือ), S (ใต้), E (ตะวันออก), W (ตะวันตก)
 - ระยะทาง: จำนวนช่องที่เดิน (จำนวนเต็มบวก)
 - ทั้งนี้ชุดคำสั่งการเดินทางเรือจะไม่พาเรือออกนอกพิกัดที่เป็นไปได้ (จะไม่พาไป row/column ที่เป็นลบ)

ตัวอย่าง: N 16 W 10 หมายถึง เดินไปทางเหนือ 16 ช่อง แล้วเดินไปทางตะวันตก 10 ช่อง

กฎการเดินทาง:

- เรือทั้งสองลำเดินพร้อมกันที่ละช่องแต่ละช่องใช้เวลา **1 หน่วยเวลา**
- ถ้าเรือลำใดเดินตามคำสั่งหมดแล้ว ให้หยุดรออยู่ที่เดิมในขณะที่อีกลำเดินต่อ



แผนที่สมบัติ

- สมบัติอยู่ที่พิกัดคงที่และมีมูลค่าเป็นจำนวนเต็ม
- เรือจะเก็บสมบัติเองโดยอัตโนมัติเมื่อเรือแล่นบนพิกัดนั้น
- สมบัติในแต่ละพิกัดจะถูกเก็บได้เพียงครั้งเดียว เก็บซ้ำไม่ได้
- กรณีเรือทั้งสองลำมาถึงพิกัดสมบัติพร้อมกันจะต้องแบ่งมูลค่ากันคนละครึ่งโดยให้หารแบบปัดเศษทิ้ง

ข้อมูลเข้า

อ่านข้อมูลต่อไปนี้จาก Standard Input

```
command
r1 c1
k1
dir1,1 dist1,1 dir1,2 dist1,2 ... dir1,k1 dist1,k1
r2 c2
k2
dir2,1 dist2,1 dir2,2 dist2,2 ... dir2,k2 dist2,k2
t
rtre1 ctre1 v1
rtre2 ctre2 v2
⋮
rtret ctret vt
```

โดยที่:

- *command* สามารถเป็นได้ 2 ค่าคือ L หรือ V
- $r_1 \ c_1$ = row และ column เริ่มต้นของเรือ A
- k_1 = จำนวนคำสั่งของเรือ A
- แต่ละคู่ $dir_{1,j} \ dist_{1,j}$ = คำสั่งเดินเรือ (ทิศทาง, ระยะทาง) ของเรือ A
- $r_2 \ c_2$ = row และ column เริ่มต้นของเรือ B
- k_2 = จำนวนคำสั่งของเรือ B
- แต่ละคู่ $dir_{2,j} \ dist_{2,j}$ = คำสั่งเดินเรือของเรือ B

- t = จำนวนพิกัดสมมติ (ไม่มีพิกัดซ้ำ)
- สำหรับแต่ละพิกัดสมมติ:
 - r_{tre_i} = row ของพิกัดสมมติ
 - c_{tre_i} = column ของพิกัดสมมติ
 - v_i = มูลค่าของสมมติ (จำนวนเต็ม)

ตัวอย่างข้อมูลเข้า

```
V
2 2
6
E 9 S 8 E 8 N 2 E 4 S 7
17 15
7
W 7 N 7 E 3 S 3 E 17 S 7 W 8
10
2 10 7
8 22 20
13 23 1000
21 5 2000
10 11 20000
2 7 5000
20 22 30
1 30 8
5 20 200000
17 13 4000
```

ขอบเขตข้อมูล

- $command \in \{L, V\}$
- $0 \leq r_1, c_1, r_2, c_2 \leq 100$
- $1 \leq k_1, k_2 \leq 100$
- $1 \leq dist_{i,j} \leq 30$
- $dir_{i,j} \in \{N, S, E, W\}$

- $1 \leq t \leq 20$
- $0 \leq r_{tre_i}, c_{tre_i} \leq 100$
- $1 \leq v_i \leq 10^{17}$

ข้อมูลส่งออก

มี 2 บรรทัด:

หาก *command* มีค่า L จะต้องแสดงค่า (Subtask 1: 20 คะแนน)

*row*₁ *col*₁
*row*₂ *col*₂

และหาก *command* มีค่า V จะต้องแสดงค่า (Subtask 2 - 5: 80 คะแนน)

*row*₁ *col*₁ *value*₁
*row*₂ *col*₂ *value*₂

โดยที่:

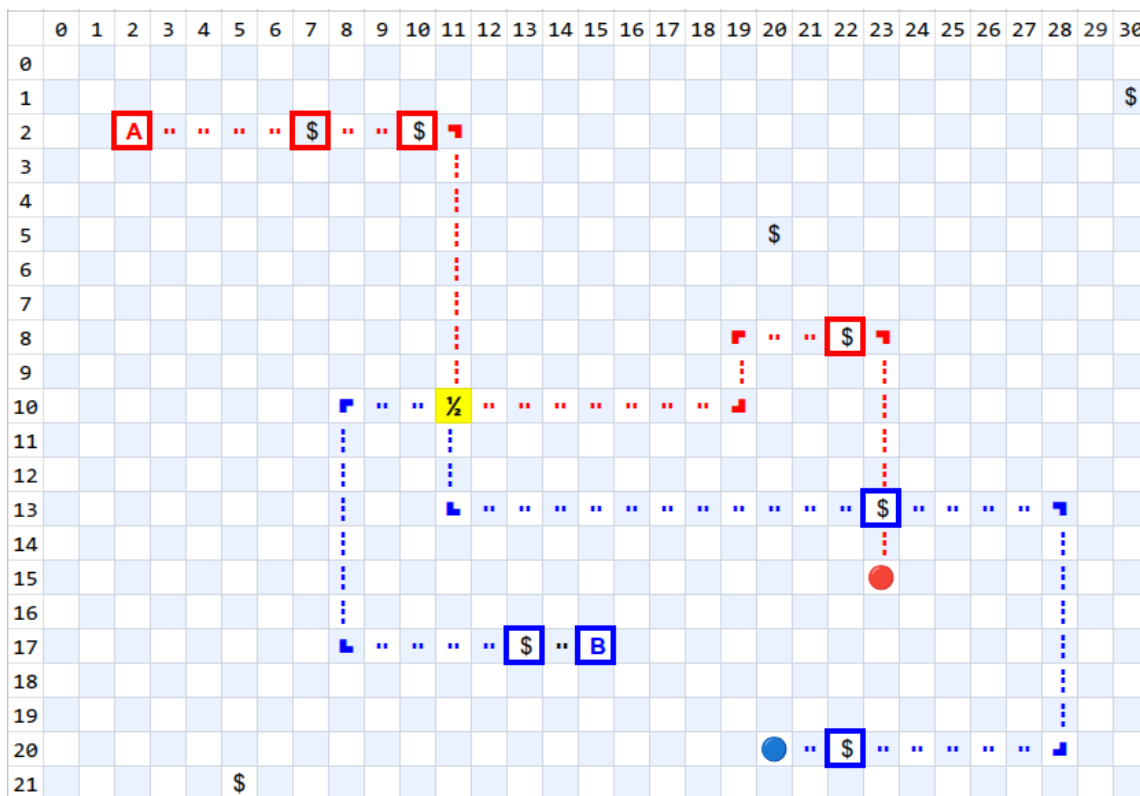
- (row_1, col_1) = ตำแหน่งสุดท้ายของเรือ A
- $value_1$ = มูลค่ารวมที่เรือ A เก็บได้
- (row_2, col_2) = ตำแหน่งสุดท้ายของเรือ B
- $value_2$ = มูลค่ารวมที่เรือ B เก็บได้

ตัวอย่างข้อมูลส่งออก

15 23
20 20

หรือ

15 23 15027
20 20 15030



คำอธิบายตัวอย่าง

จากข้อมูลตัวอย่าง

- เรือ A ผ่านพิกัดสมมติที่ $(2, 7) \rightarrow (2, 10) \rightarrow (10, 11) \rightarrow (8, 22) \rightarrow (13, 23)$
- เรือ B ผ่านพิกัดสมมติที่ $(17, 7) \rightarrow (10, 11) \rightarrow (13, 23) \rightarrow (20, 22)$
- การเก็บสมบัติเป็นไปดังนี้
 - เรือ A จะเก็บสมบัติได้จากพิกัด $(2, 7)$, $(2, 10)$ และ $(8, 22)$ ทั้งหมดเนื่องจากเป็นลำเดียวที่ผ่าน
 - เรือ B จะเก็บสมบัติได้จากพิกัด $(17, 13)$ และ $(20, 22)$ ทั้งหมดเนื่องจากเป็นลำเดียวที่ผ่าน
 - จุดที่ทับซ้อนจุดแรกคือ $(10, 11)$ เนื่องจากเรือทั้งสองลำมาถึงในเวลาเดียวกันทำให้ต้องแบ่งมูลค่าสมบัติไปลำละครึ่ง
 - จุดที่ทับซ้อนจุดที่สองคือ $(13, 23)$ เนื่องจากเรือ B มาถึงก่อนจึงได้สมบัติที่พิกัดนี้ไปทั้งหมด