

LISA DODOCOMMUNICATION 2.0



Pic : Lisa di perpustakaan knight of favonius

by: Maulana Ikhsan (Ichan)

Time Limit: 1 s

Memory Limit: 256 MB

Description

Lisa is developing the Dodocommunication Device into version 2.0. This device now uses Vision resonance as an antenna to transmit conversation signals between two people. Here are the topology and how the Dodocommunication 2.0 network works:

- Local Network (One Region): Vision users in a region are connected in a chain forming a circle (Circular Ring). Signals can propagate in both directions (left or right) to find the shortest route between Vision users.
- Global Network (Inter-Region): If the target is in another region, the signal will be sent through a waypoint, the "Tree of Life", in the origin region.
- Irminsul Transmission: The Trees of Life from all over Teyvat are directly connected to Irminsul, forming a circle (Circular Ring). The transmission mechanism on the global network (Irminsul) is the same as the transmission mechanism in the local region.
- Every time a signal changes hands (transmission/hop), its strength will weaken. The signal has a timeout limit of K seconds for each call.

Since you have a lot of free time, simulate this Dodocommunication 2.0 system!

Constraint

- $1 \leq N \leq 10$
- $1 \leq T \leq 100$
- $0 \leq V \leq 1000$
- $1 \leq V_i \leq 100$
- $1 \leq K \leq 10000$

Input Format

- The first line contains an integer N representing the number of regions.
- The next N inputs each contain 2 lines with the following rules:
 1. The first line contains a string R representing the region name.
 2. The second line contains a string V which is a list of vision holder names, separated by spaces and ending with "end".
- An integer T representing the number of testcases.
- The next T inputs each contain 1 line with the following rules:
{ "origin" "destination" K } where K is the timeout limit for that call.

Output Format

For each testcase, output one of these possibilities:

- If connection time $\leq K$, output "Connected on [connection time] second".
- If connection time $> K$, output "Signal lost".
- If the signal origin or destination is **not** in the database, output "Signal lost".

Sample Input 1

4

Mondstadt

Amber Kaeya Diluc Venti Barbara Eula end

Liyue

Xiangling Zhongli Keqing Ningguang Ganyu Zibai end

Inazuma

Ayaka Thoma Yoimiya Raiden Yae end

Sumeru

Nahida Alhaitham Cyno Nilou Tighnari end

3

Amber Zhongli 3

Kaeya Barbara 5

Zhongli Zibai 6

Sample Output 1

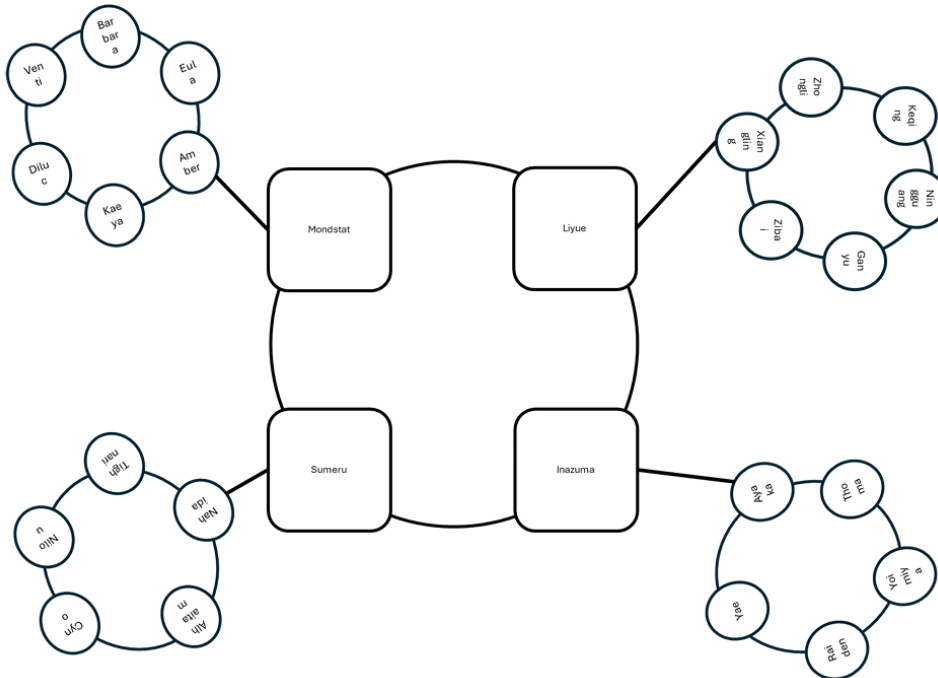
Signal lost

Connected on 3 second

Connected on 2 second

Explanation of Sample 1

Based on the topology image provided



(sorry ges probset nya gabisa gambar jadi jelek dikit)

Its is known that:

1. Route (Amber → Zhongli)
Amber → Mondstat(tree) → Liyue(tree) → Xiangling → Zhongli. There is a journey of 4 hops, so it will timeout if $K = 3$.
2. Route (kaeya → Barbara)
Kaeya → Diluc → Venti → Barbara. There is a journey of 3 hops, so it will not timeout.
3. Route (Zhongli → Zibai)
Zhongli → Xiangling → Zibai. There is a journey of 2 hops, so it will not timeout.(note that the journey can be done in both directions)

Hint

Here is hints using sample 1 as a reference

1. The orientation of the region placement (clockwise/counter-clockwise) is not important, because no matter the rotation, it will not change the route taken.
2. The same applies to the orientation of the vision holder placement, so it doesn't matter whether Kaeya is on the right or left of Amber, because they are circular so the path length is definitely the same.
3. The order of region placement refers to the input where Mondstadt is placed first, then followed by Liyue next to it (free to be right/left).
4. The same applies to vision holders, where the placement order refers to the input. For example, for Sumeru it is `Nahida` → `Alhaitham` → `Cyno` → `Nilou` → `Tighnari` → `Nahida` (back to head), where the head is directly connected to the region.
5. You can create a linked list containing linked lists, or 2 linked lists to solve it.

LISA DODOCOMMUNICATION 2.0



Pic : Lisa di perpustakaan knight of favonius

by: Maulana Ikhsan (Ichan)

Time Limit: 1 s

Memory Limit: 256 MB

Deskripsi

Lisa sedang mengembangkan Dodocommunication Device menjadi versi 2.0. Alat ini sekarang menggunakan resonansi Vision sebagai antena pengantar sinyal percakapan antar dua orang. Berikut merupakan topologi dan cara kerja jaringan dodocommunication 2.0:

- Jaringan Lokal (Satu Region): Para pengguna Vision di suatu region terhubung secara berantai membentuk lingkaran (Circular Ring). Sinyal dapat menjalar ke dua arah (kiri atau kanan) untuk mencari rute terpendek antar pengguna Vision.
- Jaringan Global (Antar Region): Jika target berada di region lain, sinyal akan dikirim melewati waypoint yaitu "Tree of Life" di region asal.
- Transmisi Irmisul: Tree of Life dari seluruh penjuru Teyvat terhubung langsung dengan Irmisul, membentuk lingkaran (Circular Ring). Cara kerja transmisi pada jaringan global(irmissul) sama seperti cara kerja transmisi pada local region.
- Setiap kali sinyal berpindah tangan (transmisi/hop), kekuatannya akan melemah. Sinyal memiliki batas timeout selama K detik untuk setiap panggilan.

Karena kamu sangatlah gabut, maka simulasilah sistem Dodocommunication 2.0 ini!

Batasan

- $1 \leq N \leq 10$
- $1 \leq T \leq 100$
- $0 \leq V \leq 1000$
- $1 \leq V_i \leq 100$
- $1 \leq K \leq 10000$

Format Masukan

- Baris pertama berisi bilangan bulat **N** yang merupakan jumlah region
- N input berikutnya masing masing berisi 2 baris dengan ketentuan:
 1. Baris pertama berisi string **R** yang merupakan nama region.
 2. Baris kedua berisi string **V** yang merupakan list nama pemegang vision, dipisahkan oleh spasi dan diakhiri oleh "end".
- Sebuah bilangan bulat **T** yang merupakan jumlah testcases.
- T input berikutnya masing masing berisi 1 baris dengan ketentuan:
{ "asal sinyal" "tujuan" **K** } dimana **K** merupakan batas timeout yang dimiliki panggilan tersebut.

Format Keluaran

Untuk setiap testcases yang ada, output salah satu dari kemungkinan ini:

- Jika connection time $\leq K$, output "Connected on [connection time] second".
- Jika connection time $> K$, output "Signal lost".
- Jika asal sinyal atau tujuan **tidak ada** pada database, output "Signal lost".

Contoh Masukan 1

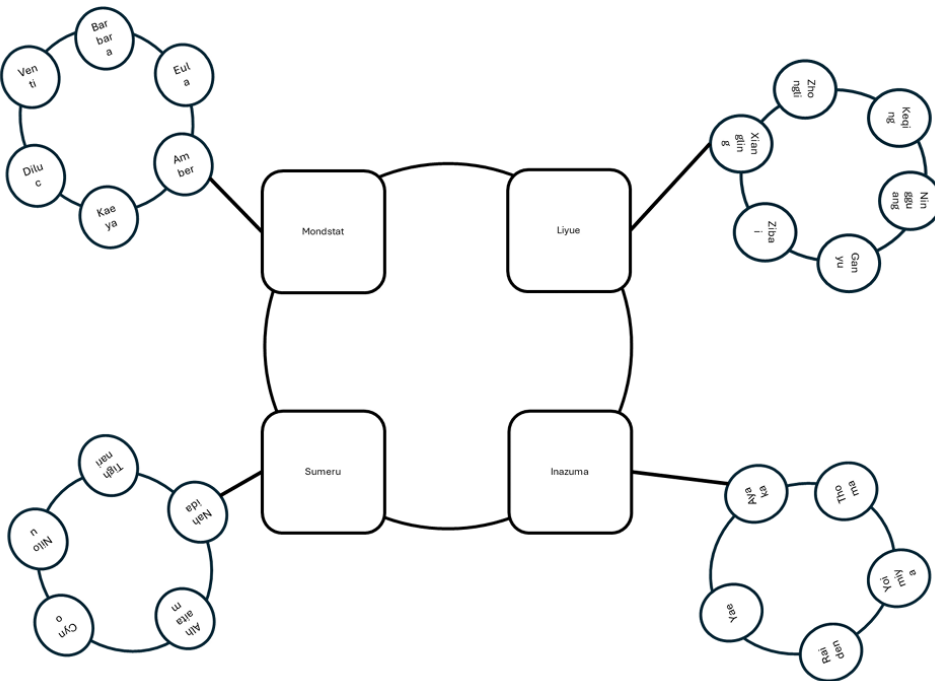
4
Mondstadt
Amber Kaeya Diluc Venti Barbara Eula end
Liyue
Xiangling Zhongli Keqing Ningguang Ganyu Zibai end
Inazuma
Ayaka Thoma Yoimiya Raiden Yae end
Sumeru
Nahida Alhaitham Cyno Nilou Tighnari end
3
Amber Zhongli 3
Kaeya Barbara 5
Zhongli Zibai 6

Contoh Keluaran 1

Signal lost
Connected on 3 second
Connected on 2 second

Penjelasan Contoh 1

Berikut merupakan gambaran topologi dari input



(sorry ges probset nya gabisa gambar jadi jelek dikit)

Perhatikan gambar diatas, dapat dilihat dan diketahui bahwa:

1. Rute (Amber → Zhongli)
Amber → Mondstat(tree) → Liyue(tree) → Xiangling → Zhongli. Terdapat perjalanan sejauh 4 hops sehingga akan timeout jika $K = 3$.
2. Rute (kaeya → Barbara)
Kaeya → Diluc → Venti → Barbara. Terdapat perjalanan sejauh 3 hops maka tidak timeout.
3. Rute (Zhongli → Zibai)
Zhongli → Xiangling → Zibai. Terdapat perjalanan sejauh 2 hops maka tidak timeout.(perhatikan bahwa perjalanan dapat dilakukan secara dua arah)

Hint

Berikut merupakan hint dengan contoh 1 sebagai acuan

1. Orientasi penempatan region(clockwise/counter-clockwise) tidaklah penting, karena bagaimana pun rotasinya tidak akan merubah rute yang akan ditempuh.
2. Hal yang sama juga berlaku untuk orientasi penempatan vision holder, jadi tidak penting apakah kaeya disebelah kanan ataupun kiri dari amber, karena mereka circular jadi pasti sama untuk panjang jalan nya.
3. Urutan penempatan region mengacu pada input dimana mondstat ditempatkan pertama, lalu dilanjut dengan liyue disebelahnya(bebas kanan/kiri).
4. Hal yang sama juga berlaku untuk vision holder, dimana urutan penempatan mengacu pada input, misal untuk sumeru maka (Nahida → Alhaitam → Cyno → Nilou → Tighnari → Nahida(back to head) dimana head lah yang terkoneksi secara langsung dengan region.
5. Bisa bikin linkedlist berisi linkedlist / 2 buah linkedlist untuk penyelesaiannya.
6. Approved solveable by "Rully Sulaiman, S.Kom., M.Kom."