Network Nexus Company

The Reply Code Masters Team

12 March 2025

1. Problem statement

Network Nexus is a dynamic technology company using a large **network infrastructure** to support the daily operations committed by employees and services. With a complex and interconnected structure, the company is composed of different types of nodes that perform vital functions for the proper functioning of the network.

Every entity in the network is exposed to a certain level of **risk of infection** by viruses or cyber attacks. The Network Nexus infrastructure is divided into three main categories of elements:

- PC: these devices are used by employees to perform daily tasks such as data management, communication and access to sites. The risk associated is significantly high due to human error and unpredictable behavior.
- Key Services: these elements are fundamental to support critical tasks such as server management, security and data storage. The risk associated is minimal due to adherence to high quality standards.
- Generic Services: these elements support secondary tasks such as e-mail management, network monitoring and backup services. The risk associated

is intermediate due to less stringent security requirements.

In order to ensure the reliability of the entire system, the Security Team calculated the **estimated risk** of each node. You can do better! When a node connects to an adjacent node with a higher risk, its risk increases indirectly. For this reason, it is essential to determine the **real risk** of each node.

How to calculate the real risk for each node:

- The node A has N adjacent nodes. For each of N couples of nodes, risk value is calculated. The maximum value calculated among the N couples of nodes represents the real risk of the node A.
- The risk value is calculated as the average between the estimated risk of the node A and the real risk of the adjacent node, rounded up.
- If the real risk of an adjacent node is not available, the estimated risk of the adjacent node is used.
- The real risk of a node can never be lower than its estimated risk. If the calculated risk value is lower, the real risk is set equal to the estimated risk.

This process needs to be repeated for all nodes, ensuring a complete and up-to-date risk assessment. The final output will contain the real risk of all the key services of the network.

2. Input format

The input file is a regular ASCII text file. Each line of the input file ends with a single $\backslash n$ character (UNIX-style). If a line contains multiple data, each value is separated by a single space character.

The first row of the input file will be composed of an integer number C, representing the number of test cases to be solved. The following rows represent for each test case C_i in ascending order:

- The integer number N_i, indicating the number of nodes provided
- N lines containing the key informations of each node:
  - I_i, indicating the identifier of the node
  - R_e, indicating the estimated risk of the node
  - $I_1 \ldots I_m$, the list of IDs of the connected nodes

The type of node is identified by the first character of the identifier I_i:

- the key services start with the character k ($k1$, $k2$, ka, kg, kz)
- the generic services start with the character g ($g1$, $g2$, ga, gg, gz)
- the personal computer start with the character c ($c1$, $c2$, ca, cg, cz)

3. Output format

The output file must be a regular ASCII text file. Each line of the output file must end with a single $\backslash n$ character (UNIX-style). The rows represent for each test case, in ascending order:

- a string indicating the test case number, in the format $Case\ \#C_i$:
- an integer N_k indicating the number of key services
- a line with the ID of the key services followed by their real risk

For example: Case $\#C_i$: N_k I_0 R_{r0} I_1 R_{r1} ... I_{N_k-1} R_{rN_k-1}

Where C_i is the test case number, from 1 to C, N_k is the number of key services provided, and I_0 R_{r0} I_1 R_{r1} $\ldots$ I_{N_k-1} R_{rN_k-1} are the key services with identifier I_i followed by the calculated real risk R_{ri}.

4. Constraints

- Risk values are integer numbers: $R_e, R_r \leq 10$
- **Input 1** : $C = 1$, $N \leq 15$
- **Input 2** : $C = 10$, $N \leq 1000$
- **Input 3** : $C = 5$, $N \leq 1500$
- **Input 4** : $C = 7$, $N \leq 1500$

5. Example

5.1. Input file example

```
1
5
k0 0 g0
k1 0 g0
g0 2 k0 k1 c0 c1
c0 5 g0
c1 2 g0
```

In this example, players have 1 network with 5 nodes. The first node $k0$ has an expected risk of 0 and is linked to the node $g0$. The second node $k1$ has an expected risk of 0 and is linked to the node $g0$. And so on.

5.2. Output file example

```
Case #1: 2 k0 2 k1 2
```

In this example, players individuated 2 key services, respectively $k0$ with a real risk equal to 2 and $k1$ with a real risk equal to 2.

6. CTF recall

Alex's habit of storing passwords in a locked file might seem secure, but what if the system logs reveal hidden clues? Attackers often extract sensitive data by analyzing patterns in logs—just like in that CTF challenge. Maybe those injection attempts hold more than just garbage input. Time to investigate.