1167 – Dangerous Tunnels

Somewhere in the world, there are two tribes separated by mountains. The two tribes are named Kulolo and Gulolo, respectively, where Kulolo is at a higher altitude and Gulolo is at a lower altitude. Due to the limitation of geography, Gulolo has fewer resources than Kulolo. In order to transport resources from Kulolo to Gulolo efficiently, several tunnels were built inside the mountains between these two tribes. There are also some rest stations built for people to take a break during the transportation in the tunnels. More specifically, each terminal of a tunnel is one of Kulolo, Gulolo, or a rest station.

The structure of those tunnels is not stable. A dangerous degree has been estimated for each tunnel, due to its stability, in advance. A tunnel with a higher dangerous degree is considered to be more dangerous; that is, it is more probably to collapse. Kinglolo, the chief of Kulolo, would like to select some paths through the tunnels to Gulolo with little risk. In Kinglolo's opinion, the dangerous degree of a path is equal to the maximum dangerous degree of the tunnels in the path; and the dangerous degree of a set of paths is equal to the maximum dangerous degree of the paths in it. For example, consider Figure 1. The dangerous degrees of $\mathbf{P_1}$, $\mathbf{P_2}$ and $\mathbf{P_3}$ are, respectively 3, 5, and 6. And, the dangerous degree of $\mathbf{\{P_2, P_3\}}$ is 6.

Since all tunnels are narrow, a limited quantity of resources can be transported along a path in one day. Therefore, every day, depending on the amount of resources needed to be transported, a different number, say **k**, of paths is required. Moreover, to avoid congestion, these **k** selected paths cannot pass any rest station in common. For example, in Figure 1, $\mathbf{P_2}$ and $\mathbf{P_3}$ can be selected at the same time; however, $\mathbf{P_1}$ and $\mathbf{P_2}$ cannot be selected at the same time, since they both pass $\mathbf{r_2}$.

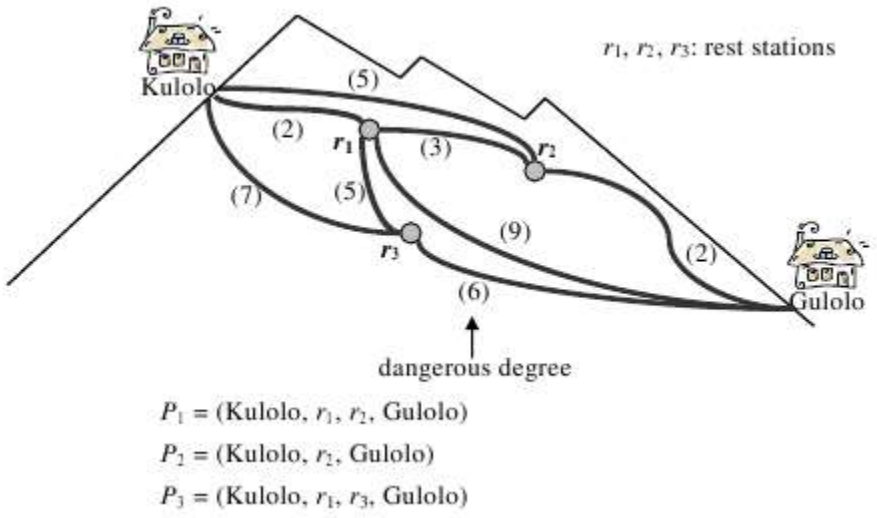

Figure 1: An example.

Kulolo has a higher altitude than all rest stations while Gulolo has a lower altitude than all rest stations. Kinglolo is a thoughtful chief. It is ensured that the altitudes of the rest stations on each

selected path are non-increasing, so that the path is more suitable for transportation. For example, in Figure 1, the path **(Kulolo, r_3, r_1, Gulolo)** will never be selected, since r_1 is higher than r_3.

People in Kulolo believe that Kinglolo is the most brilliant man in the world, since he always selects a set of **k** paths that is as little dangerous as possible (i.e., the maximum dangerous degree of the selected paths is minimized). Now, given the data of the constructed tunnels, you are asked to find **k** paths that Kinglolo may select. In summary, the **k** selected paths, if exist, should satisfy the following:

1. all paths are from Kulolo to Gulolo,
2. no two paths pass the same rest station,
3. the altitudes of the rest stations on each path are non-increasing, and
4. the maximum dangerous degree of the paths is minimized.

For simplicity, only the maximum dangerous degree of the selected paths should be reported.

Input

Input starts with an integer **T (≤ 30)**, denoting the number of test cases.

Each case starts with a blank line and a positive integer **n (0 < n ≤ 100)** which indicates that there are **n** rest stations $r_1, r_2, \ldots, r_n$. For ease of description, Kulolo and Gulolo are denoted by r_0 and r_{n+1}, respectively. We assume that r_i is higher than r_j for any **(0 ≤ i < j ≤ n+1)**. The second line of each case contains a positive integer **t (> 0)** that specifies the number of tunnels. Each of the following **t** lines contains three integers **p, q, d (0 ≤ p ≤ n+1, 0 ≤ q ≤ n+1, p ≠ q, 1 ≤ d ≤ 100000)** separated by white space, which indicate there is a tunnel with dangerous degree **d** connecting r_p and r_q . Then, a line containing a positive integer **k (1 ≤ k ≤ 10)** is provided, which is the number of paths that should be selected. You can assume that there is at most one tunnel between any two rest stations.

Output

For each case, print the case number and the maximum dangerous degree of the **k** paths that Kinglolo may select. If the solution does not exist, print **'no solution'**.

Sample Input	Output for Sample Input
4 2 4 0 1 3 1 3 12 2 0 10 2 3 5 1 1 2 0 1 5 1 2 6 2 3 2 0 1 5 3 4 7 1 3 6 0 1 8 0 2 12 0 3 15 3 1 9 3 4 8 2 4 12 2	Case 1: 10 Case 2: no solution Case 3: no solution Case 4: 12