1094 – Farthest Nodes in a Tree

Given a tree (a connected graph with no cycles), you have to find the farthest nodes in the tree. The edges of the tree are weighted and undirected. That means you have to find two nodes in the tree whose distance is maximum amongst all nodes.

Input

Input starts with an integer **T (≤ 10)**, denoting the number of test cases.

Each case starts with an integer **n (2 ≤ n ≤ 30000)** denoting the total number of nodes in the tree. The nodes are numbered from **0** to **n-1**. Each of the next **n-1** lines will contain three integers **u v w (0 ≤ u, v < n, u ≠ v, 1 ≤ w ≤ 10000)** denoting that node **u** and **v** are connected by an edge whose weight is **w**. You can assume that the input will form a valid tree.

Output

For each case, print the case number and the maximum distance.

Sample Input	Output for Sample Input
2 4 0 1 20 1 2 30 2 3 50 5 0 2 20 2 1 10 0 3 29 0 4 50	Case 1: 100 Case 2: 80

Notes

Dataset is huge, use faster i/o methods.