Problem C. 0 Tree

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

We have a tree $\langle V, E\rangle$ that consists of n vertices numbered from 1 to n. Each vertex $i \in V$ has weight a_i. Each bidirectional edge $e = \langle u, v\rangle \in E$ has weight b_e. Here, a_i are non-negative integers, and b_e are integers.

You can perform at most $4n$ operations. For each operation, select two vertices X and Y, and a non-negative integer W. Consider the shortest path from X to Y (a path is shortest if the number of edges k in it is minimum possible). Let this path consist of $k+1$ vertices $(v_0, v_1, v_2, \ldots, v_k)$ where $v_0 = X$, $v_k = Y$, and for $0 \le i < k$, the edges $e_i = \langle v_i, v_{i+1}\rangle \in E$. The operation changes the weights as follows:

$$a_X \leftarrow a_X \bigoplus W; \quad a_Y \leftarrow a_Y \bigoplus W; \quad b_{e_i} \leftarrow b_{e_i} + (-1)^i \cdot W \text{ for } 0 \le i < k.$$

Here, $\bigoplus$ denotes the bitwise XOR operation. We can notice that, if $X = Y$, nothing will change.

You need to decide whether it is possible to make all a_i and all b_e equal to 0. If it is possible, find a way to do so.

Input

The first line contains an integer T ($1 \le T \le 250$), the number of test cases. Then T test cases follow.

The first line of each test case contains a single integer n ($1 \le n \le 10^4$), the number of vertices.

The second line contains n non-negative integers a_i ($0 \le a_i < 2^{30}$), the weight on each vertex.

Then $n-1$ lines follow, each of them contains three integers u_j, v_j, w_j ($1 \le u_j, v_j \le n$, $-10^9 \le w_j \le 10^9$), representing an edge between vertices u_j and v_j with weight w_j. It is guaranteed that the given edges form a tree.

It is guaranteed that $\sum n \le 10^5$.

Output

For each test case, output "`YES`" on the first line if you can make all a_i and all b_e equal to 0 with no more than $4n$ operations. Output "`NO`" otherwise.

If you can make all weights equal to 0, output your solution in the following $k+1$ ($0 \le k \le 4n$) lines as follows.

On the next line, print an integer k: the number of operations you make.

Then print k lines, each line containing three integers X, Y, and W ($1 \le X, Y \le n$, $0 \le W \le 10^{14}$), representing one operation.

If there are several possible solutions, print any one of them.

Example

standard input	standard output
3 1 0 2 2 3 1 2 -2 3 5 4 1 1 2 -5 2 3 -5	YES 0 NO YES 3 1 3 5 2 3 7 2 3 3