Niceness of the Arrays

Problem Code: **NICARRAY**

All submissions for this problem are available.###Read problems statements Mandarin , Bengali , Hindi , Russian and Vietnamese as well.

Let's define the *niceness* of a sequence of positive integers $X_1, X_2, \ldots, X_N$ X1,X2,...,XN as the sum of greatest common divisors of all pairs of its elements, i.e.

$$\sum_{i=1}^{N} \sum_{j=i+1}^{N} \gcd(X_i, X_j) \ .$$

∑i=1N∑j=i+1Ngcd(Xi,Xj).

For example, the niceness of the sequence $[1, 2, 2]$[1,2,2] is $\gcd(1, 2) + \gcd(1, 2) + \gcd(2, 2) = 4$ gcd(1,2)+gcd(1,2)+gcd(2,2)=4.

You are given a sequence $A_1, A_2, \ldots, A_N$ A1,A2,...,AN; each of its elements is either a positive integer or missing.

Consider all possible ways to replace each missing element of AA by a positive integer (not necessarily the same for each element) such that the sum of all elements is equal to S S. Your task is to find the total niceness of all resulting sequences, i.e. compute the niceness of each possible resulting sequence and sum up all these values. Since the answer may be very large, compute it modulo $10^9 + 7$ 109+7.

Input

- The first line of the input contains a single integer T T denoting the number of test cases. The description of T T test cases follows.
- The first line of each test case contains two space-separated integers N N and S S.
- The second line contains N N space-separated integers $A_1, A_2, \ldots, A_N$ A1,A2,...,AN. Missing elements in this sequence are denoted by -1 −1.

Output

For each test case, print a single line containing one integer — the total niceness modulo $10^9 + 7$ 109+7.

Constraints

- $1 \le T \le 20$ 1≤T≤20
- $1 \le N, S \le 50$ 1≤N,S≤50
- $1 \le A_i \le 50$ 1≤Ai≤50 or $A_i = -1$ Ai=−1 for each valid i

Subtasks

Subtask #1 (30 points):

- $1 \le N, S \le 18$ 1≤N,S≤18
- $1 \le A_i \le 18$ 1≤Ai≤18 or $A_i = -1$ Ai=−1 for each valid ii

Subtask #2 (70 points): original constraints

Example Input

```
3
3 3
1 1 -1
4 8
1 -1 -1 3
3 10
-1 -1 -1
```

Example Output

```
3
23
150
```

Explanation

Example case 1: There is only one possible way to fill in the missing element; the resulting sequence is $[1, 1, 1]$[1,1,1]. Its niceness is $3$3.

Example case 2: There is only three possible ways to fill in the missing elements; the resulting sequences are $[1, 1, 3, 3]$[1,1,3,3], $[1, 3, 1, 3]$[1,3,1,3], and $[1, 2, 2, 3]$[1,2,2,3]. The sum of their niceness is

Country Tour

Problem Code: **CTOUR**

All submissions for this problem are available.Ishank lives in a country in which there are N N cities and $N-1$ N−1 roads. All the cities are connected via these roads. Each city has been assigned a unique number from 1 to N N. The country can be assumed as a tree, with nodes representing the cities and edges representing the roads. The tree is rooted at 1.Every Time, when a traveler through a road, he will either gain some amount or has to pay some amount.

Abhineet is a traveler and wishes to travel to various cities in this country. There's a law in the country for travelers, according to which, when a traveler moves from the city AA to city BB, where city AA and BB are connected by a road then the traveler is either paid or has to pay the amount of money equal to profit or loss respectively. When he moves from AA to BB, he hires a special kind of vehicle which can reverse its direction at most once. Reversing the direction means earlier the vehicle is going towards the root, then away from the root or vice versa. Abhineet is analyzing his trip and therefore gave QQ queries to his friend, Ishank, a great coder. In every query, he gives two cities AA and BB. Ishank has to calculate the maximum amount he can gain (if he cannot gain, then the minimum amount he will lose) if he goes from the city AA to city BB.

Input:

-The first line of the input contains a two space-separated integers N and Q.

-The next N-1 line contains 3 space-separated integers Xi and Yi and Zi denoting that cities Xi and Yi are connected by a road which gives profit Zi (Negative Zi represents loss).

-The next Q contains 2 space-separated integers A and B denoting two cities.

Output:

Print a single line corresponding to each query — the maximum amount he can gain (if he cannot gain, then the minimum amount he will lose with negative sign) if he goes from city A to city B.

Constraints

- $2 \leq N \leq 10^5$ 2≤N≤105
- $1 \leq Q \leq 10^5$ 1≤Q≤105
- $1 \leq Xi, Y i, A, B \leq N$ 1≤Xi,Yi,A,B≤N
- $abs(Zi) \leq 10^9$ abs(Zi)≤109

Sample Input:

```
9 5
1 2 8
1 3 -9
2 4 1
2 5 -6
3 6 7
3 7 6
6 8 3
6 9 4
1 2
2 7
4 3
3 2
8 9
```

Sample Output:

```
10
5
0
-1
21
```

EXPLANATION:

In the first query, he goes from 1 to 2, 2 to 4, takes a turn and go to 2. Therefore profit=8+1+1=10.

Knight Chess

Problem Code: **KCHESS**

All submissions for this problem are available.Abhineet the Chess master of NIT Kurukshetra got bored of 8x8 chess board and invented the new variation of Chess, the one on an infinite chess board. There is only a white king and N N black knights. The white king has to avoid checkmate as long as it can.

A situation is given. Determine if white king is in checkmate or not. The white king is in checkmate if and only if it is in check and it is not able to move to any of its neighboring positions which is not in check.

Input:

- The first line will contain T T, number of test cases. Then the test cases follow.
- The first line of each test case contains a single integer N N.
- The next N line contains 2 space-separated integers Xi Xi and Y iYi denoting the position of knights.
- The next line contains 2 space-separated integers A A and B B denoting the position of king

---

Output:

Print a single line corresponding to each test case — “YES” (without quotes) – if the king is in checkmate and “NO” (without quotes) – if the king is not in checkmate.

Constraints

- $1 \leq T \leq 5$ 1≤T≤5
- $1 \leq N \leq 10^5$ 1≤N≤105
- $-10^9 \leq Xi, Y\,i, A, B \leq 10^9$ −109≤Xi,Yi,A,B≤109

Sample Input:

```
2
6
1 0
0 2
2 5
4 4
5 0
6 2
3 2
4
5 6
3 5
2 8
-8 -8
```

Sample Output:

YES

NO

EXPLANATION:

In the first test case, the king is in cannot move to any valid position In second test case, the king can easily make a valid move.