Currency Exchange

Summer is coming soon and Lea wants to travel the world. Since she was a kid, she always dreamed of visiting the Great Temples of Templonia, and this summer her dream shall come true. As Templonians pay with their own currency, the Column, Lea has to go to a bank to exchange currencies. The Column is very rarely exchanged, so she decides to go to the National Bank, which has almost all the currencies of the world available. While looking at the current exchange rates, she may just have discovered a loophole in the system: Changing a currency via several exchanges may leave her with more money than just changing to the desired currency in one step.

Soon, Lea realized that the optimal sequence of exchanges is found by multiplying exchange ratios. Luckily, she remembered a grade school course she had taken on calculus, and knows that $\log(a \cdot b) = \log a + \log b$. Moreover, $\log(a \cdot b)$ is minimal if and only if $a \cdot b$ is minimal. This way, Lea can sum the logarithms of the exchange rates and find the optimal sequence. Can you help Lea write a program to find the best way to change her money into Columns?

Input

The first line of the input contains an integer t. t test cases follow, each of them separated by a blank line.

Each test case starts with a line containing two integers n and m, separated by a space. n denotes the number of currencies (w.l.o.g. the currencies are labeled 1 to n), m denotes the number of possible exchanges between currencies. m lines follow. The i-th line consists of two integers a_i, b_i, and a double c_i, separated by spaces, which means that the i-th exchange gives the rate c_i for changing the currency a_i into the currency b_i, i.e., one can change c_i units of currency a_i into one unit of currency b_i. Note that this does not imply that Lea can change money back from currency b_i to a_i.

The doubles c_i are given with a dot as the decimal symbol.

Lea's current money is given in currency 1, and the Column is represented by currency n.

Output

For each test case, output one line containing "Case #i: x" where i is its number, starting at 1, and x is one of the three following answers: If Lea can make an infinite amount of money in any currency, then x is `Jackpot`. Otherwise x is either the best (smallest) exchange rate achieved via a sequence of exchanges; or `impossible` if there is neither a possible exchange between the two currencies nor a way to make infinite money. Your output must have an absolute or relative error of at most 10^{-6}.

Constraints

- $1 \leq t \leq 20$
- $1 \leq n \leq 500$
- $0 \leq m \leq 5000$
- $1 \leq a_i, b_i \leq n$ for all $1 \leq i \leq m$
- $a_i \neq b_i$ for all $1 \leq i \leq m$
- $0 < c_i < 25$ for all $1 \leq i \leq m$
- For any (ordered) pair of currencies there exists at most one exchange rate.

Sample Input 1	Sample Output 1
3 4 5 1 2 0.6 1 4 2.0 2 3 0.4 3 1 4.5 4 3 0.4 4 5 1 2 0.6 1 4 2.0 2 3 0.4 3 1 3.0 4 3 0.4 4 5 1 2 1 1 3 1 3 2 0.5 2 4 1 3 4 1	Case #1: 2.000000 Case #2: Jackpot Case #3: 0.500000

Sample Input 2	Sample Output 2
3 8 10 8 2 2.253667193183237 7 3 21.06270642254913 4 1 16.54304400691424 6 2 11.417274520023643 3 8 0.4842040541716247 3 5 24.89220955129525 6 5 3.9879932153224855 1 3 3.3862989246293607 4 3 18.6557548269335 5 2 4.044574982952042 6 4 4 1 5.49865346857874 2 3 1.041528946247336 4 6 15.198096869720814 2 6 20.72926928096376 5 8 4 2 15.234936910338007 3 2 2.783840758662068 2 3 4.34452037823547 1 5 5.149609531472857 5 1 21.634342229523323 5 3 21.628749390916028 5 2 22.244249973539482 5 4 17.64965379097069	Case #1: 1.639660 Case #2: impossible Case #3: 5.149610