ACM International Collegiate Programming Contest, Egyptian Collegiate Programming Contest Parallel 2017

A. Assessments

time limit per test: 6 seconds
memory limit per test: 1024 megabytes
input: assessment.in
output: standard output

As the World Cup approaches, FIFA is holding a worldwide assessment of players. In each assessment, we have a line of N distinct players, the players are standing in the order $0, 1, ..., N - 1$. The assessment is carried out in k rounds, in each round the players run for a long time. When they finish the round, it sometimes happens (with probability p) that exactly one pair of adjacent players end up swapping their positions; this pair of adjacent players occurs randomly with equal chance. FIFA wants your help finding out the probability that the final state after the k rounds will have the players x and y swapped.

Input

The first line of the input contains a single integer $1 \le T \le 100$ the number of test cases. Each test case consists of one line that contains 5 space separated numbers N, p, x, y, k; where $1 \le N \le 50$ is the number of players, $0 \le x, y < N$ are the numbers of the players of the query, $0 \le k \le 3000$ is the number of rounds of the assessment, and p is a floating point number $0.0 \le p \le 1.0$.

Output

For each test case, output a single line displaying the case number and the answer probability rounded to exactly 5 decimal places.

Example

input
3 4 1.0 1 1 2 4 0.0 0 2 1 2 0.5 0 1 1
output
Case 1: 0.33333 Case 2: 0.00000 Case 3: 0.50000

B. Breaking the Curse

time limit per test: 15 seconds
memory limit per test: 1024 megabytes
input: curse.in
output: standard output

Egypt has finally qualified to the World Cup 2018 after 28 years. It was almost a curse that has been broken. The main factor of this curse was the goal of Abdelghany in the World Cup 1990. Abdelghany has been talking over and over again about this goal that Egyptians started to regret that they have even qualified to that World Cup. Now, since Egypt has made it to the World Cup, lots have said that the curse is over, but of course Abdelghany disagrees. The curse is about someone scoring for Egypt there after him, not only participating. Hence, all Egyptians are praying day and night for any player to score in the 2018 World Cup.

Of course all the hopes are for Egypt's star Salah to score in the World Cup. Egyptians are interested in knowing the probability that Salah will score in the World Cup and end this curse. Egyptian Scientists decided to calculate this probability for the benefit of the country and the sanity of its people. They encapsulated Salah's traits, like DNA, personality and quality in a string s_1 and did the same for Abdelghany in a string s_2. Now, they will study the similarity of these two strings in order to calculate the probability of Salah scoring in the World Cup, like Abdelghany. They are making some experiments initially in order to be sure and they want your help, by answering q queries each consisting of two integers L and R; in each query you should find the number of intervals $[i, j]$ (inclusive) where $L \le i \le j \le R$ and $s_1[i...j]$ is a substring of s_2.

Input

The first line of the input contains a single integer $1 \le T \le 50$ the number of test cases. Each test case begins with two lines, the first line is s_1 and the second line is s_2. The two lines are followed by a line containing a single integer q the number of queries, then followed by q lines each containing two space separated integers L, R the values of the query; where the strings s_1, s_2 consist of lower case English characters, $1 \le |s_1|, |s_2|, q \le 75 \cdot 10^3$.

Output

For each test case output a line displaying the case number, followed by q lines each containing a single integer which is the answer to the corresponding query.

Example

input
2 bbc abb 4 1 3 1 2 2 3 3 3 hadafkaselalaam kas 1 1 10

output
Case 1: 3 3 1 0 Case 2: 8

Statement is not available.

D. Dream Team

time limit per test: 8 seconds
memory limit per test: 1024 megabytes
input: dream.in
output: standard output

Before the World Cup, we would like to form the ultimate dream team. The team that will win against all teams. We have a big group of the top N players in the world, where each player has a number representing his skill level. Two players i and j with skill levels a_i and a_j have a compatibility of passing the ball between each other; this compatibility level is measured by the greatest common divisor of their skill levels (i.e. $compatibility(i, j) = gcd(a_i, a_j)$). We would like to decide a strategy that connects all the players of the dream team with a tree of connections between the players. If two players are directly connected in the chosen strategy, then they will pass the ball between each others during the matches, with the compatibility level between them. The compatibility of the strategy is the sum of the compatibility levels of its connections. What is the maximum total compatibility of the chosen strategy that connects all players?

Input

The first line of the input contains a single integer $1 \le T \le 50$ the number of test cases. Each test case consists of one line that contains $N + 1$ space separated integers; the first integer is N, followed by the skill levels $a_1, ..., a_n$; where $1 \le N \le 10^5$ and $1 \le a_i \le 10^5$ for all i.

Output

For each test case, print a single line displaying the case number and the maximum compatibility of the dream team's strategy.

Example

input
2 2 3 6 5 5 6 7 10 21
output
Case 1: 3 Case 2: 17

Note

In the second test case we connect the players 3 - 5 with compatibility 7, players 1 - 4 with compatibility 5, players 2 - 5 with compatibility 3 and players 2 - 4 with compatibility 2; therefore the total compatibility of the dream team's strategy is 17.

E. Evaluations

time limit per test: 14 seconds
memory limit per test: 1024 megabytes
input: evaluations.in
output: standard output

For World Cup marketing purposes, FIFA wants to evaluate the compatibility of different players. Some players have connections that affect their compatibility scores, like being from the same country or playing in the same team. These connections are well studied, and for some pairs of players, an integer value is assigned based on these connections. After such studies, FIFA found out that the set of players with connections form an undirected tree. In more details, players are the nodes of the tree and edges connect players with a connection such that the edge's weight is equal to the value of this connection. It is guaranteed that these edges will form a tree. The compatibility score of players u, v is:

- If there is an edge (u, v) in the tree, then it is the weight of that edge.
- Otherwise, it is the weight of the path between u and v, such that the weight of a path is the product of the weights of its edges.

Given the tree of players along with edges' weights, FIFA is interested in counting the number of distinct unordered pairs of players u, v such that their compatibility score is a product of exactly two distinct primes.

Input

The first line of the input contains a single integer $1 \le T \le 100$ the number of test cases. Each test case consists of n lines, the first line containing a single integer n. Each of the remaining $n - 1$ lines consists of 3 space separated integers $x\ y\ w$; each of those denotes an edge between the nodes x and y with weight w; where $1 \le x, y \le n \le 10^5$ and $1 \le w \le 10^5$.

Output

For each test case output a single line displaying the case number and the count.

Example

input
1 5 1 2 2 2 3 1 1 4 3 1 5 6
output
Case 1: 3

Note

An unordered pair is a set of two distinct elements. $\{a, b\} = \{b, a\}$.

Statement is not available.

Statement is not available.

H. Half Nice Years

time limit per test: 4 seconds
memory limit per test: 1024 megabytes
input: halfnice.in
output: standard output

Salah is really optimistic about the qualification of the Egyptian football team for the world cup 2018. He was thinking deeply about why it didn't happen often that the Egyptian team qualified to the world cup, so he decided to call 2018 a half-nice year. He comes up with the hypothesis that the Egyptian team will always qualify to the world cup only if it is held in a half-nice year. In order to test his hypothesis we need to find the largest half-nice number in a given range $[A, B]$. He says a given number X is half-nice if its decimal representation can be divided into two non-zero halves(of the same length) and the two halves have a non-trivial common divisor (there's a number $d > 1$ that divides both parts). In case a number X have odd number of digits ($2r + 1$), then the first half will have the first r digits in the decimal representation of X (from the left), and the second half will contain the remaining $r + 1$ digits in the decimal representation of X.

Input

The first line of the input consists of a single integer $1 \le T \le 10^5$ the number of test cases. Each test case consists of one line containing 2 space separated integers A, B, where $10 \le A \le B \le 10^{13}$.

Output

For each test case output a single line displaying the case number, then followed by either a single integer X, which is the largest half-nice number in the range $[A, B]$ or 'impossible' if there are no such numbers.

Example

input
3 20 21 10 2019 4000 4005

output
Case 1: impossible Case 2: 2018 Case 3: 4005

Note

- The number 20122 is a half-nice number (hopefully the Egyptian team qualifies that year), because it's two halves 20 and 122 have a non-trivial common divisor which is 2.
- The number 2000 is not a half-nice number because the second half is zero.
- In the third, test case 4005 is half-nice because 40 and 5 share the non-trivial common divisor 5. It is allowed for the second half to have trailing zeros.

I. Important matches

time limit per test: 15 seconds
memory limit per test: 1024 megabytes
input: important.in
output: standard output

In the qualification rounds of the World Cup, there were two disjoint sets of teams $S_1 = [1, ..., N]$ and $S_2 = [1, ..., M]$. Given a matrix X that decides the importance of the match between team $i \in S_1$ against team $j \in S_2$ by the value $X_{i,j}$. S_1 represents the rows of the matrix X, and S_2 represents the columns of the matrix X. We would like to answer Q queries, where each decides the importance of all the matches played by all the teams from the subgroup $[A, ..., C]$ from S_1 against all teams from the subgroup $[B, ..., D]$ from S_2, such that $1 \le A \le C \le N$, $1 \le B \le D \le M$. The importance of a set of matches between two subgroups is determined by the median importance of all the matches between the teams from the first subgroup against all the teams from the second group, namely the median of the submatrix $X_{A... C, B... D}$.

Input

The first line of the input contains a single integer $1 \le T \le 100$ the number of test cases. Each test case begins with a line containing 3 spaces separated integers N, M, Q. The importance matrix X is given by the next N lines each containing M space separated integers. The remaining Q lines, each contains 4 space separated integers A, B, C, D of the query; where $1 \le N, M \le 200$, $1 \le Q \le 10000$, and for the given importance matrix X, $1 \le X_{i,j} \le 2000$ for all i, j.

Output

For each test case output a line displaying the case number, followed by Q lines each containing a single integer, the importance of the matches of the corresponding query.

Example

input
1 3 4 2 1 2 3 1 2 1 1 4 7 8 9 3 1 1 1 1 1 2 3 4

output
Case 1: 1 3

Note

- The median of a set of numbers is the middle element in the sorted list of the given set. If the set has two middle elements, then we choose the second one. For example the median of {2, 1, 3} is 2 and the median of {4, 2, 3, 1} is 3.
- The second query asks for the median of the submatrix that is excluding the first column. This submatrix contains the elements {2, 3, 1, 1, 1, 4, 8, 9, 3}, which has the sorted form {1, 1, 1, 2, 3, 3, 4, 7, 8, 9}, therefore the median is 3.

Statement is not available.

K. Katryoshka

time limit per test: 1 second
memory limit per test: 1024 megabytes
input: katryoshka.in
output: standard output

The Egyptian football team will be in Russia for the World Cup, of course they all would like to buy souvenirs for their families. Luckily they met the king of souvenirs of the famous Russian souvenir Matryoshka; the king makes his masterpiece Katryoshka using different kinds of pieces: wooden eyes, wooden mouths, and wooden bodies. He can form a nice Katryoshka using one of the following ways:

- Two eyes and a body.
- Two eyes, mouth and a body.
- Eye, mouth and a body.

If the king has n eyes, m mouths and k bodies, what is the largest number of Katryoshkas that can be made by the king?

Input

The first line of the input contains a single integer $1 \le T \le 100$ the number of test cases. Each test case consists of a single line containing 3 space separated integers 'n m k' the number of eyes, mouths and bodies respectively; where $0 \le n, m, k \le 10^8$.

Output

For each test case output a line displaying the case number and a single integer which is the largest number of nice Katryoshka pieces.

Example

input
4 1 2 3 0 11 2 14 21 23 90 24 89
output
Case 1: 1 Case 2: 0 Case 3: 14 Case 4: 57

L. Lazy ERCD

time limit per test: 1 second
memory limit per test: 1024 megabytes
input: lazy.in
output: standard output

FIFA changed the style of the World Cup. There are no group rounds anymore, therefore the World Cup will be a pure knockout competition. In a knockout competition, each match that is played between two teams, the losing team is knocked-out of the competition and will not play again, until there's exactly one winner (we are not concerned about other positions, only the first place).

Having N teams in the World Cup, our ERCD wants to know how many matches he needs to watch (he will really watch all matches). He is lazy to count that number, so he needs you to write a program that calculates it.

Input
The first line of the input contains a single integer $1 \leq T \leq 100$ the number of test cases. Each test case consists of 1 line, containing a single integer N, the number of teams; where $1 \leq N \leq 100$.

Output
For each test case output a single line displaying the case number, followed by the number of matches the ERCD will watch.

Example

input
3 2 87 4

output
Case 1: 1 Case 2: 86 Case 3: 3