icpc International Collegiate Programming Contest



The 2026 ICPC North America Pacific Northwest Regional Contest Division 2

Official Problem Set

Google Cloud
Google DeepMind
icpc diamond global sponsor

JETBRAINS
icpc global sponsor programming tools

Jane Street
icpc multi-regional titanium sponsor

NATIONAL SECURITY AGENCY
north america platinum sponsor

north america institutional sponsor

Probook
icpc sponsor

This page is intentionally left blank.

A: Bangaranga

Time Limit: 1 second

Dara is preparing her latest song, Bangaranga. The song will be played at a volume of d decibels, but Dara is unhappy with the volume and wants to test out various adjustments to the volume.

Decibels work on a logarithmic scale, so an increase from d decibels to $d+10$ decibels corresponds to being 10 times more intense. In general, an increase from d decibels to $d+10k$ decibels corresponds to being 10^k times more intense. Symmetrically, a decrease from d decibels to $d-10$ decibels corresponds to being 10 times less intense. In general, a decrease from d decibels to $d-10k$ decibels corresponds to being 10^k times less intense.

Dara's adjustments take two forms:

- `BANGA k`: Take the current song and make it k times more intense.
- `RANGA k`: Take the current song and make it k times less intense.

These changes are cumulative. After each change, Dara wants to know the volume of the song.

Input

The first line of input contains a single integer T ($1 \leq T \leq 50$), representing the number of test cases.

The descriptions of T test cases follow. Each test case starts with a line containing two integers, d ($1 \leq d \leq 200$) and q ($1 \leq q \leq 50$), the initial volume of the song in decibels and the number of adjustments, respectively.

The next q lines each describe one adjustment, in the order Dara makes them. Each line is of the form either `BANGA k` or `RANGA k`, where k ($10 \leq k \leq 10^9$) is an integer power of 10. It can be shown that under these constraints, the volume will always be an integer number of decibels.

Output

For each test case, for each adjustment, output the volume of the song in decibels after that adjustment.

Sample Input 1	Sample Output 1
2 37 5 BANGA 10 BANGA 10000 RANGA 1000 BANGA 100 RANGA 10 1 1 RANGA 10	47 87 57 77 67 -9

B: Cowtrain

Time Limit: 1 second

Nick has been caught sending strange messages in Telegram and claims his keyboard has been hacked. Specifically, he claims that some keys, when pressed, map to different letters from the ones that appear on his keyboard.

If Nick's keyboard is actually hacked, two things are true:

- Every time Nick presses a given key, the same letter appears.
- No two different keys produce the same letter.

Given the string Nick wanted to type and the string he did type, is Nick's claim possible?

Input

The first line of input contains a single integer T ($1 \leq T \leq 50$), representing the number of test cases.

The descriptions of T test cases follow. Each test case consists of two lines. The first line contains a string s ($1 \leq |s| \leq 50$) of lowercase letters, the string Nick wanted to type. The second line contains a string t ($|t| = |s|$) of lowercase letters, the string Nick typed.

The strings s and t will be different.

Output

For each test case, if it is possible for Nick's claim to be valid, output `YES`. Otherwise, output `NO`.

Sample Input 1	Sample Output 1
2 cal cow caltrain cowtrain	YES NO

This page is intentionally left blank.

C: Floor Ceil Fiasco

Time Limit: 1 second

Computers are able to exactly represent base-10 integers up to some limit with a series of 0s and 1s. However, they sometimes struggle with representing base-10 decimals, which have to be rounded off at some point, leading to some weird behavior (for example, try adding 0.1 and 0.2 and printing the result—many languages won't print the mathematically correct answer). To avoid this, many languages will default to integer operations: if both operands are integers, the output is usually an integer as well.

Integer division is problematic if the numerator isn't divisible by the denominator, though. Most languages will round down (floor) the result to the nearest integer, so the integer result may be less than the actual quotient. However, the result from rounding up (ceil) may be greater than the actual quotient.

You will be given two integers n and k. You must divide n by 2 a total of k times. On each division, you may choose to floor the result, rounding it down, or ceil it, rounding it up. Given n and k, output the number of possible resulting integers after applying any combination of k floors or ceils.

Input

The first line of input contains a single integer T ($1 \leq T \leq 50$), representing the number of test cases.

The descriptions of T test cases follow. Each test case consists of a single line containing two integers, n and k ($1 \leq n \leq 10^9$, $0 \leq k \leq 30$).

Output

For each test case, output the number of possible resulting integers after applying any combination of k floors or ceils.

Sample Input 1	Sample Output 1
3 16 3 4 3 7 0	1 2 1

This page is intentionally left blank.

D: Inside Regions

Time Limit: 1 second

Audrey is practicing her calligraphy and is experimenting with different aesthetic styles. One day, she realizes that depending on how she writes the digit 4, there can either be one enclosed region inside, or none. She starts to wonder how many enclosed regions each digit has.

Eventually, she decides that the correct way to write each of the 10 digits has the following number of enclosed regions:

Digit	Number of Enclosed Regions
0	1
1	0
2	0
3	0
4	1
5	0
6	1
7	0
8	2
9	1

Given a digit, determine how many enclosed regions it has.

Input

The first and only line of input contains a digit d ($0 \leq d \leq 9$).

Output

Output the number of enclosed regions the digit has.

Sample Input 1	Sample Output 1
4	1

Sample Input 2	Sample Output 2
7	0

Sample Input 3	Sample Output 3
8	2

E: Iron Supplementation

Time Limit: 1 second

Nick is deficient in iron, and his doctor has prescribed an iron supplementation regimen for him. Specifically, Nick has a prescription to ingest m milligrams of iron every morning for d days. Due to how iron is processed in the body, every night, p percent of the iron stored in the body disappears. Nick is so deficient in iron that he starts out with zero milligrams of iron stored.

Compute the number of milligrams of iron Nick has stored after the night of the d^{th} day.

Input

The first line of input contains a single integer T ($1 \leq T \leq 50$), representing the number of test cases.

The descriptions of T test cases follow. Each test case consists of a single line containing three integers, m, d, and p ($1 \leq m \leq 50$, $1 \leq d \leq 50$, $1 \leq p \leq 99$).

Output

For each test case, output the number of milligrams of iron stored after the night of the d^{th} day. Your answer will be considered correct if it has absolute or relative error at most 10^{-3}.

Sample Input 1	Sample Output 1
2 10 2 10 24 39 67	17.1 11.82089552238806

This page is intentionally left blank.

F: Split One

Time Limit: 1 second

Nick is walking to work. His watch treats his walking commute as an outdoor walk workout and notifies him of how long it takes him to walk a mile. Unfortunately, one day he doesn't get the notification and wants to know how long it took him to walk the first mile of his commute.

Nick starts his commute at time 0. His commute has n intermediate checkpoints. Nick is so consistent in his walking speed that between two consecutive checkpoints (and between the start of his commute and the first checkpoint), his speed is constant.

Input

The first line of input contains a single integer T ($1 \leq T \leq 50$), representing the number of test cases.

The descriptions of T test cases follow. Each test case starts with a line containing a single integer n ($1 \leq n \leq 50$), the number of checkpoints.

The next n lines each describe one checkpoint in the order Nick reaches them. Each line contains two integers, d and t ($1 \leq d \leq 10^4$, $1 \leq t \leq 10^4$), indicating that the checkpoint is d feet from the start of Nick's commute and that Nick reaches it t seconds after starting.

The values of d are strictly increasing, the values of t are strictly increasing, and the last value of d is at least 5280 (the number of feet in a mile).

Output

For each test case, output the number of seconds it took Nick to walk the first mile of his commute. Your answer will be considered correct if it has absolute error at most 1.

Sample Input 1	Sample Output 1
3 2 2640 360 5280 720 2 2640 360 5281 361 1 5281 1	720 360.999621356 0.999810642

This page is intentionally left blank.

G: Agentic Development

Time Limit: 1 second

You would like to develop a system on your computer, and you've heard that you can use an AI agent to do the work for you as long as you can break the system down into manageable tasks. Developing the system requires the successful completion of n tasks, which can be undertaken in any order, but only one at a time. For each task, you know how long the agent will take to attempt it. At the end of each attempt, exactly one of three things happens:

1. The agent successfully completes the task.
2. The agent fails to complete the task, but no harm is done.
3. The agent irreversibly corrupts your system. You immediately wipe the system clean and start over with no completed tasks.

For each task, you know the probability of success, failure, and corruption. Each attempt is independent of every other attempt. Wiping the system clean takes no time. You choose which task the agent attempts each time, and you would like to minimize the expected time taken. What is the expected time to complete all tasks and develop the system, assuming you direct the agent optimally?

Input

The first line of input contains a single integer T ($1 \le T \le 100$), representing the number of test cases.

The descriptions of T test cases follow. Each test case starts with a line containing a single integer n ($1 \le n \le 100$), the number of tasks needed to develop the system.

The next n lines each describe one task. Each line contains four integers, t, s, f, and c ($1 \le t \le 1000, 0 \le s, f, c \le 100$), indicating that each attempt at the task takes time t, and that the attempt succeeds with probability $\frac{s}{s+f+c}$, fails with probability $\frac{f}{s+f+c}$, and corrupts the system with probability $\frac{c}{s+f+c}$.

For each task, at least one of s, f, and c will be positive.

Output

For each test case, output the minimum expected time for the agent to complete all tasks and develop the system. Your answer will be considered correct if it has relative error at most 10^{-6}.

If it is impossible for the agent to complete all tasks, output `impossible`.

Sample Input 1	Sample Output 1
3 2 1 2 3 4 5 6 7 8 2 4 3 2 1 8 7 6 5 1 67 0 67 67	28 34.285714285714 impossible

H: Birds and Bread

Time Limit: 4 seconds

Kyle is a man who likes two things: birds and bread. He spends his Saturday afternoons feeding the ducks at his local community center's pond. He sits on a bench, rips off a piece of bread, and throws the piece of bread into the pond. As the piece of bread hits the pond, it creates a ripple in the shape of a circle that expands over the pond. While throwing pieces of bread, Kyle would like to figure out how many intersections there are between the different ripples on the pond.

Kyle models the pond as an infinite 2D plane. Each piece of bread thrown lands at integer coordinates (x, y). If a piece of bread lands at time t, its ripple starts as a single point. The ripple radiates outwards as a circle, with the radius growing at a rate of one unit per second. Therefore, at time $t + k$, the ripple has a radius of k and is centered at (x, y). If a piece of bread is thrown on a ripple, this counts as 1 intersection.

If several ripples intersect at the same point, every pair counts as a separate intersection at that point.

Input

The first line of input contains a single integer T $(1 \leq T \leq 200)$, representing the number of test cases.

The descriptions of T test cases follow. Each test case starts with a line containing two integers, n $(1 \leq n \leq 2 \cdot 10^3)$ and q $(1 \leq q \leq 10^5)$, where n is the number of pieces of bread Kyle throws and q is the number of times Kyle counts the intersections.

The next n lines each have three integers `t x y` $(0 \leq t \leq 10^9, -10^9 \leq x, y \leq 10^9)$, indicating that a piece of bread lands on the water at time t at coordinates (x, y). The times t for the pieces of bread are non-decreasing. Kyle will never throw two pieces of bread at the same location at the same time.

The next q lines each have a single integer t $(0 \leq t \leq 10^9)$, indicating that Kyle wants to count the number of intersections at time t. These times are strictly increasing.

The sum of n over all test cases will not exceed $2 \cdot 10^3$, and the sum of q over all test cases will not exceed 10^5.

Output

For each time that Kyle counts the intersections, output a line with the number of intersections at that time. It can be shown that the number of intersections is always finite.

Sample Input 1	Sample Output 1
2 3 1 0 -1 0 0 1 0 0 0 1 1 3 3 0 0 0 0 6 8 6 6 8 5 6 67	5 1 2 4

I: Eins, Zwei, Drei

Time Limit: 2 seconds

Have you found a language to count in that brings you back to life?

Sam has composed a song in which each integer from 1 to n is sung exactly once. Each integer is sung in one of k languages, and Sam has already decided which language each integer will be sung in.

The producers require that, for each language, the integers sung in that language appear in a specific order. However, Sam may interleave integers from different languages in any way he wishes.

The producers also prefer that smaller integers be sung before larger integers, so they define the score of a song to be the length of its longest increasing subsequence. A subsequence of the song is obtained by deleting zero or more of its integers without changing the order of the rest, and it is increasing if each integer in it is larger than the one before it.

Help Sam find an order to sing the integers that maximizes the score.

Input

The first line of input contains a single integer T ($1 \le T \le 10^4$), representing the number of test cases.

The descriptions of T test cases follow. Each test case starts with a line containing two integers, n ($1 \le n \le 2 \cdot 10^5$) and k ($1 \le k \le n$), the number of integers in the song and the number of languages, respectively.

The next k lines each describe one language. Each line starts with an integer l ($1 \le l \le n$), followed by the l integers sung in that language, in the order the producers require. Together, these k lines contain each integer from 1 to n exactly once.

The sum of n over all test cases will not exceed $2 \cdot 10^5$.

Output

For each test case, output a single line containing the integers from 1 to n in the order Sam should sing them. Integers sung in the same language must appear in the order given in the input, and the score of the song must be as large as possible.

If there are multiple such orders, any will be accepted.

Sample Explanation

For the first test case, it is possible to interleave the integers such that Sam sings the integers from 1 to 5 in sorted order. This is the only order that yields a score of 5.

For the second test case, we can show that the maximum possible score is 4. One possible longest increasing subsequence of the provided sample output can be obtained by deleting the integers 8, 7, 6, 5, and 2, leaving 1, 3, 4, and 9. Another output that would be accepted for this test case is `4 7 8 2 5 6 9 3 1`.

Sample Input 1	Sample Output 1
2 5 3 2 1 4 2 2 5 1 3 9 3 3 4 2 9 3 7 5 3 3 8 6 1	1 2 3 4 5 8 7 6 1 5 3 4 2 9

J: Pareto Optimal

Time Limit: 4 seconds

Audrey is in charge of a club called the Mathematically Curious Summer Programmers. Each person is rated on two attributes, their mathematical skill and their curiosity.

Audrey needs to send a representative to the Magical Institution of Trigonometry and wants to put her club in the best possible light. Audrey, knowing that the ratings for mathematical skill and curiosity can be imprecise, picks an integer k.

A person with mathematical skill m_i and curiosity c_i is ineligible for consideration if and only if there exists some other person with mathematical skill m_j and curiosity c_j such that at least one of the two conditions is true:

- $m_j \geq m_i + k$ and $c_j \geq c_i$.
- $m_j \geq m_i$ and $c_j \geq c_i + k$.

List all people that are eligible for consideration.

Input

The first line of input contains a single integer T ($1 \leq T \leq 10^4$), representing the number of test cases.

The descriptions of T test cases follow. Each test case starts with a line containing two integers, n ($1 \leq n \leq 2 \cdot 10^5$) and k ($1 \leq k \leq 10^9$). Audrey's club has n people.

The next n lines each contain two integers, m and c ($1 \leq m, c \leq 10^9$), the mathematical skill and curiosity of each person. These people are numbered from 1 to n in the order given. No two people in the same test case have both the same mathematical skill and the same curiosity.

The sum of n over all test cases will not exceed $2 \cdot 10^5$.

Output

For each test case, output two lines. The first line should contain an integer p, the number of people who are eligible for consideration. The second line should contain p integers in increasing order, the numbers of the people eligible for consideration. It can be shown that there will always be at least one person eligible for consideration.

Sample Input 1	Sample Output 1
3 5 1 3 3 3 2 2 3 2 2 1 1 5 2 3 3 3 2 2 3 2 2 1 1 2 1 1 2 2 1	1 1 4 1 2 3 4 2 1 2

K: Peppa's Puddles

Time Limit: 1 second

Peppa the Pig loves jumping in puddles. She sees n puddles laid out in a line, numbered from 1 to n from left to right, and she wants to jump in as many of them as possible. She can jump in each puddle at most once.

Unfortunately, jumping in a puddle may splash one or both of its adjacent puddles. If a puddle is splashed before Peppa has jumped in it, she can no longer jump in that puddle. Splashing a puddle that Peppa has already jumped in has no effect.

Help Peppa determine the maximum number of puddles she can jump in, if she chooses an optimal order to jump in them.

Input

The first line of input contains a single integer T ($1 \le T \le 10^4$), representing the number of test cases.

The descriptions of T test cases follow. Each test case starts with a line containing a single integer n ($2 \le n \le 2 \cdot 10^5$), the number of puddles.

The next line contains a binary string s_L ($|s_L| = n - 1$). The i^{th} character of s_L is `1` if jumping in puddle $i + 1$ splashes puddle i, and `0` otherwise.

The next line contains a binary string s_R ($|s_R| = n - 1$). The i^{th} character of s_R is `1` if jumping in puddle i splashes puddle $i + 1$, and `0` otherwise.

The sum of n over all test cases will not exceed $2 \cdot 10^5$.

Output

For each test case, output the maximum number of puddles that Peppa can jump in.

Sample Input 1	Sample Output 1
2 3 11 11 3 00 11	2 3

This page is intentionally left blank.

L: Televote Resolver

Time Limit: 2 seconds

Gena is watching the International Country and Pop Contest (ICPC). In the ICPC, n teams each perform a song and are awarded points by a jury and by the televote. The jury have given out all of their points, and the number of points each team got from the jury has been announced to the public. It remains to resolve the number of points given by the televote, which are kept secret until they are announced. The protocol for this is as follows:

- The teams are sorted in nondecreasing order by the number of points they got from the jury.
- The announcers iterate over the teams in that order, and announce the number of points they got from the televote.

The winner of the ICPC is the team with the largest sum of jury points and televote points. If more than one team has the largest number of points, one of those teams is picked at random to win.

In addition to the jury points, Gena knows that a total of exactly p points will be given in the televote, but does not know how many televote points any team got until it is announced. It may be possible to know exactly who has won the ICPC before the announcers have announced all the results. Formally, what is the smallest integer $c \geq 0$ such that, after hearing the results of the televote for the first c teams, Gena can uniquely identify the winner of the ICPC?

Input

The first line of input contains a single integer T ($1 \leq T \leq 10^3$), representing the number of test cases.

The descriptions of T test cases follow. Each test case starts with a line containing two integers, n ($2 \leq n \leq 2 \cdot 10^5$) and p ($1 \leq p \leq 10^{18}$).

The next n lines each contain two integers, j and t ($0 \leq j \leq 10^{18}, 0 \leq t \leq p$), the number of points the team earned from the jury and from the televote, respectively. The teams are given in the order in which the announcers announce them.

The values of j are nondecreasing, and the values of t sum to exactly p.

The sum of n over all test cases will not exceed $2 \cdot 10^5$.

Output

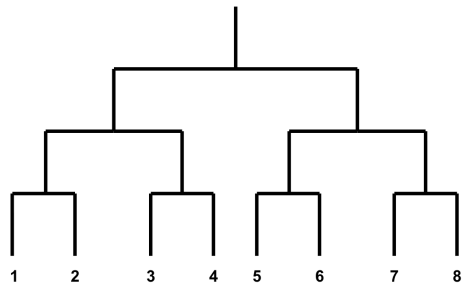
For each test case, output the smallest integer c such that, after announcing the results of the televote for the first c teams, the winner of the ICPC is uniquely determined. If the winner is not uniquely determined after all the televote results are announced, output -1.

Sample Input 1	Sample Output 1
3 2 1 0 0 67 1 2 100 1 67 2 33 2 2 1 1 1 1	0 1 -1

M: Tournament Rigging

Time Limit: 1 second

There is a single-elimination tournament where players compete in 1-on-1 matches. The full bracket is known in advance. All of the players are sufficiently well known that for each pair of players, you know which player would win in a fair 1-on-1 match. However, you also know that everyone has their own price, and thus you know how much of a bribe each player would require in order to intentionally lose a match.



Before any given match, you can either do nothing or give a bribe to one of the participants. If you do nothing, the match will be fair. Otherwise, the person who was bribed will lose.

What is the minimum total amount of money you would need to invest in bribes in order to ensure that player 1 wins the entire tournament?

Input

The first line of input contains a single integer T ($1 \leq T \leq 1024$), representing the number of test cases.

The descriptions of T test cases follow. Each test case starts with a line containing a single integer k ($1 \leq k \leq 11$), denoting that there are 2^k players, numbered from 1 to 2^k. The bracket is set up in order of the players, so the first round matches player 1 against player 2, player 3 against player 4, etc., and the next round matches the winner among (1, 2) against the winner among (3, 4), etc.

The next 2^k lines each describe one player, in order from player 1 to player 2^k. Each line contains an integer b ($1 \leq b \leq 10^6$) and a string s ($|s| = 2^k$) of 0s and 1s, where b is the cost to bribe this player and the i^{th} character of s is 1 if and only if this player would win against player i in a fair match.

For each player i, the i^{th} character of player i's string is 0. For each pair of distinct players i and j, exactly one of the following is true: the j^{th} character of player i's string is 1, or the i^{th} character of player j's string is 1.

The sum of 2^k over all test cases will not exceed 2048.

Output

For each test case, output the minimum total cost of bribes needed to ensure that player 1 wins the entire tournament.

Sample Input 1	Sample Output 1
3 1 3 00 5 10 2 3 0110 1 0011 4 0001 1 1000 3 9 01010101 6 00111111 4 10011111 8 00001111 100 10000110 2 00000011 3 10000001 5 00001000	5 0 7