Contents

Notes

- Many problems have large input file sizes, so we suggest using fast I/O. For example:
  - In Java, use `BufferedReader` and `PrintWriter`.
  - In C/C++, use `scanf` and `printf`.
- The problems are solvable in C++, and Java, but the same is not guaranteed for Python due to its slowness.
- Good luck!

Problem A
Statistricks

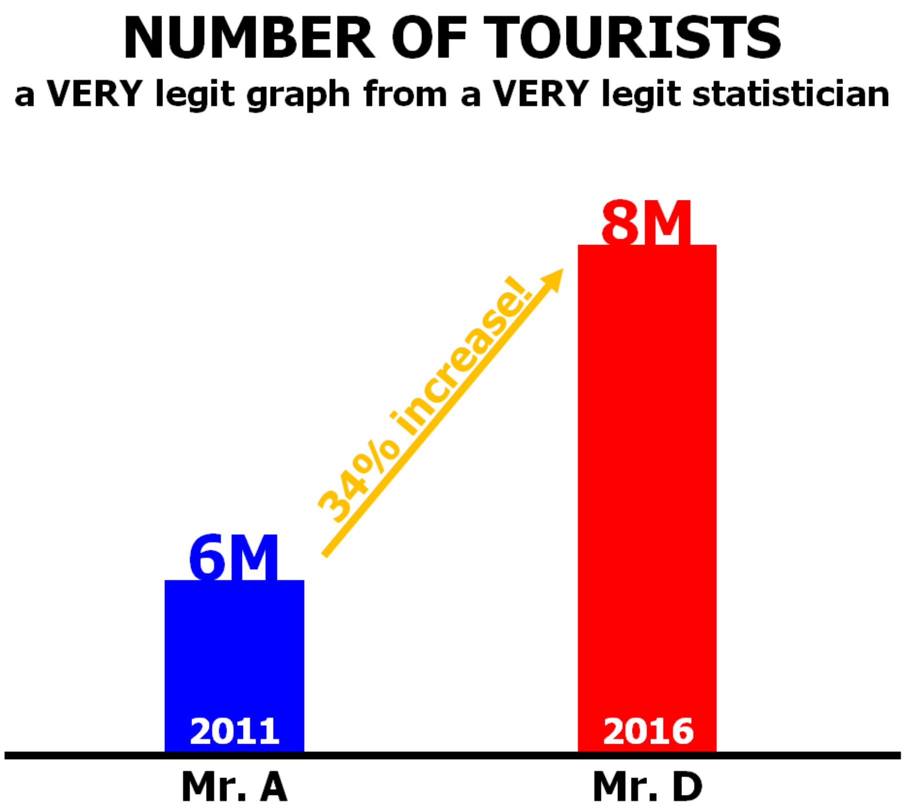
Time Limit: 3 seconds

Rommel, one of the die-hard supporters of Mr. D, wants to make publicity materials highlighting his boss's achievements. In particular, he wants to insinuate that the management of Mr. D is better than that of his predecessor, Mr. A.

For a given criterion, say the amount of tourists who flocked the country, yearly data is available from $2010-2016$. Suppose the corresponding data for the year y is x_y. Rommel wants to state something of the form:

"During the year m, Mr. A's administration attracted x_m million tourists to the country. In 2016, Mr. D's administration attracted x_{2016} million tourists – an n percent increase compared to the x_m million tourists in Mr. A's administration."

where n is the correct percentage rounded **up** to the nearest integer.



Rommel can choose m to be any year between 2010 and 2015 inclusive since those are the years of Mr. A's term as the boss. To make the best statement possible, he chooses m such that the corresponding integer n is maximized.

Given $x_{2010}, \ldots, x_{2016}$, find the highest possible value of n. However, if n is not positive, Rommel would choose to claim that the data is ~~bias~~ biased instead of accepting the data given as it is.

Input Format

The first line of input contains t, the number of test cases.

The next t lines will contain seven space-separated positive integers $x_{2010}, x_{2011}, \ldots, x_{2016}$ in that order.

Constraints

$1 \leq t \leq 3 \cdot 10^4$
$1 \leq x_i \leq 100$

Output Format

For each test case, output `n% INCREASE!` where n is an integer, as discussed in the problem statement. However, if n is not positive, output `DATA IS BIAS!` instead.

Sample Input	Sample Output
2 7 6 7 8 9 9 8 7 7 7 7 7 7 1	34% INCREASE! DATA IS BIAS!

Problem B

Fashionista

Time Limit: 3 seconds

Golda is a fashionista and she has a collection of n different fashion accessories, indexed from 0 to $n-1$, each of which she can either choose to wear or not wear on any single day. This means that she wears a set S of accessories from her collection. We refer to this set S as an *outfit*.

As a fashionista who is also into combinatorics, she knows that she has 2^n different outfits. She indexes them in the following way: If $0 \le y < 2^n$ and we write $y = \sum_{k=0}^{n-1} x_k \cdot 2^k$ where $x_k \in \{0,1\}$, then outfit y is the outfit wherein she wears accessory k if and only if $x_k = 1$.

However, choosing which accessories to wear takes a lot of her time. Because of this, she decides to construct a schedule that she will use starting tomorrow until the end of time. In particular, her schedule is a list of the 2^n outfits in some order. In the following, we detail the constraints she imposes on the soon-to-be-decided schedule.

- She intends to literally cycle through the outfits in the same order. This means that once she wears the last outfit on the list, she shall wear the first outfit on the list on the day after.
- She does not want to repeat outfits as much as possible. This means that if you look at her outfits within any 2^n consecutive days, there should be no two days when she wears the same outfit.
- She does not want to be noticed too much. This is why she decides that her outfit on any two consecutive days must be exactly the same except for one particular accessory. More formally, if S and T are the sets of accessories she wears in two consecutive days, then either $T = S \cup \{k\}$ for some $k \notin S$ or $S = T \cup \{k\}$ for some $k \notin T$.
- She requires that outfit a is the ith outfit on the list and outfit b is the jth outfit on the list.

Your task is to make her a schedule, or determine if it's impossible.

Input Format

The first line of input contains t, the number of test cases.

Each test case consists of a single line containing five integers n, i, a, j and b.

Constraints

$1 \le t \le 2^{16}$
$1 \le n \le 16$
$0 \le a, b < 2^n$
$0 < i, j \le 2^n$
The sum of 2^n across all test cases is $\le 2^{20}$

Output Format

For each test case, print a line containing either `YES` or `NO` denoting whether it's possible to find a schedule that satisfies all the constraints above. In addition, if you print `YES`, print a second line containing 2^n integers denoting the schedule, separated by single spaces. Any valid schedule will be accepted.

Sample Input	Sample Output
2 2 2 2 4 1 4 7 1 6 15	YES 0 2 3 1 NO

Problem C

Bananas in Pajamas

Time Limit: 3 seconds

B1 and B2 are anthropomorphic bananas. In case you're not sure what that *big* word means, it is a word used to describe a long curved fruit which grows in clusters and has soft pulpy flesh and yellow skin when ripe.

Whenever they're both in pajamas, they somehow have some telepathic powers. For example, their conversations lately have been about integers. They converse about a certain nonnegative integer n that both of them are thinking:

B2: Are you thinking of what I'm thinking B1?

B1: I think I am B2!

B2: Are you thinking of an integer which has exactly y_2 2's when written in base y?

B1: Yes, and it has y_1 1's when written in that same base.

B2: Indeed! Also, when written in base x, the integer has exactly x_2 2's.

B1: True. In fact, it has exactly x_1 1's when written in that same base.

Given x_1, x_2, y_1, y_2, find nonnegative integer values of x, y, n such that both bananas were telling the truth during the entire conversation.

For the purposes of this task, we shall require that the bases x and y be at least 3 and at most 10. Also, when written in any of the two bases, the nonnegative integer n must have at most 700 digits.

Note that such values of x, y, n may not exist. If this is the case, you must say so.

Input Format

The first line of input contains t, the number of test cases.

Each test case consists of a single line containing four space-separated integers x_1, x_2, y_1, y_2.

Constraints

$1 \leq t \leq 7000$
$0 \leq x_1, x_2, y_1, y_2 \leq 100$

Output Format

The first line of output for each test case contains either YES or NO, denoting whether it's possible or not. If it's possible, print two more lines describing x, y and n:

- The first line contains x, a space, and the base x representation of n.
- The second line contains y, a space, and the base y representation of n.

The representation should not contain a leading zero unless $n = 0$.

Sample Input	Sample Output
3 3 2 1 0 2 0 1 2 3 4 2 3	YES 3 1200121 4 103033 YES 4 133130 5 31022 YES 7 400250362211021 10 2718281828459

Explanation

In the first example, we have $x_1 = 3$, $x_2 = 2$, $y_1 = 1$ and $y_2 = 0$. One possible answer is $x = 3$, $y = 4$ and $n = 1231$ (in decimal), since n written in base 3 is 1200121 which contains exactly 3 1's and 2 2's, and n written in base 4 is 103033 which contains exactly 1 1's and 0 2's.

In the second example, we have $x_1 = 2$, $x_2 = 0$, $y_1 = 1$ and $y_2 = 2$. One possible answer is $x = 4$, $y = 5$ and $n = 2012$ (in decimal). You can verify that this is a correct answer.

Problem D
Weird Keyboard
Time Limit: 3 seconds

Bogarrt recently got a job which involves typing out handwritten documents on the computer. Since he is a good typist, he came to this job overconfident.

Just like a normal keyboard, pressing a key labeled x on his work keyboard would add the string x to the digital document he was working with. His overconfidence suddenly waned when he realized that his work keyboard was not the conventional type of keyboard. It had keys, but all of them were labeled by strings which were not necessarily length 1! Moreover, all of these keyboards did not have any way to delete text or move the cursor around.

For example, one of the keyboards had 4 keys – `AR`, `BOG`, `SCHMOGART`, `T`. Bogarrt quickly realizes that he cannot even type his name with this keyboard. The closest he can do is `BOGART` or `BOGARART`. But what does it mean for strings to be close?

The *edit distance* between two strings s and t can be obtained by finding the least number of operations needed to modify s to become t, using only these three basic operations:

- **Adding a letter.** For example, `BOGART` can be modified to become `BOGARRT`.
- **Deleting a letter.** For example, `BOGARART` can be modified to become `BOGARRT`.
- **Replacing a letter.** For example, `BOGAHRT` can be modified to become `BOGARRT`.

For example, the edit distance between `BOGARART` and `BOGART` is 2. As another example, the edit distance between `BOAT` and `BETS` is 3 (`BOAT` $\rightarrow$ `BOET` $\rightarrow$ `BET` $\rightarrow$ `BETS`).

We denote by $d(s,t)$ the edit distance between strings s and t.

Let t be the string that Bogarrt's company wants him to type. Your task is to find the closest string to t that you can type, in terms of the edit distance. More formally, let S be the set of all strings which can be formed by pressing a nonnegative number of keys in Bogarrt's keyboard. You need to find the minimum value of $d(s,t)$ over all $s \in S$.

Input Format

The first line of input contains c, the number of test cases.

Each test case will consist of exactly three lines. The first line of each test case will contain an integer k, the number of keys that Bogarrt can press. The second line will contain k space-separated strings $s_1, s_2, \ldots, s_k$ containing only uppercase letters. The third line will contain a string t of uppercase letters. This will be the string that Bogarrt needs to type.

Constraints

$1 \le c \le 4$
$1 \le k \le 4000$
$1 \le |t|, |s_i| \le 4000$
$1 \le |s_1| + |s_2| + \ldots + |s_k| \le 4000$

Output Format

For each test case, output a single line containing a single integer denoting the answer for that test case.

Sample Input

```
2
4
AR BOG SCHMOGART T
BOGARRT
4
BA KITA MA TI
MASISISIBAKITA
```

Sample Output

```
1
3
```

Problem E

Agents of Shield

Time Limit: 4 seconds

S.H.I.E.L.D. has p different headquarters around the country. Their main purpose is to be an agency for counter-terrorism. This means that they hire spies and transport them around different places. However, they also pick-up and deliver special bath soaps as a side business.

Delivering special bath soap is not very simple. For example, let's say a building x may have a bath soap that is to be delivered to building y. An S.H.I.E.L.D. agent must first go to building x to pick-up the special bath soap there. Once an agent picks up the bath soap, he/she must be the one to deliver it to building y. That is, the agent cannot pass it to other agents. An agent can also choose to pick-up and deliver several bath soaps at the same time, and in any order. Also, an agent is able to carry an unlimited number of soaps.

Agents must start their day from any of the p headquarters. From there, an agent is only allowed to use company-approved transportation. The two types of transportation are as follows:

- Transportation between two particular buildings a_i and b_i is allowed under a company car. The cost of using a car to go from a_i to b_i (or vice-versa) is c_i monies.
- Transportation between any two S.H.I.E.L.D. headquarters is free via helicopter and does not cost anything.

There are m pairs of buildings in which the agents of S.H.I.E.L.D. can use a company car. They will be specified in the input. Moreover, it is assured that all n buildings that will be in the input are reachable.

An S.H.I.E.L.D. agent need not return to a headquarters upon finishing the delivery of the bath soaps.

Given k special bath soaps, their respective buildings of origin, and their intended destinations, what is the minimum number of monies that must be spent in order to deliver them all? Note that one can hire as many S.H.I.E.L.D. agents as needed.

Input Format

The first line of input contains four space-separated integers n, m, k, and p. The buildings will be numbered from 1 to n.

The next m lines describe the pairs of buildings in which the agents can use a company car. Specifically, the ith following line contains three space-separated integers denoting a_i, b_i, and c_i.

The next line contains p space-separated integers $h_1, h_2, \ldots, h_p$ denoting the building numbers of the p headquarters.

The next k lines describe the bath soaps that need to be delivered. In particular, the ith following line contains two space-separated integers s_i and t_i denoting that the ith bath soap starts at building s_i and must be delivered to building t_i.

Constraints

$3 \leq n \leq 60000$
$2 \leq m \leq 120000$
$1 \leq k \leq 11$
$1 \leq p \leq n$
$1 \leq a_i, b_i, h_i, s_i, t_i \leq n$
The h_is are distinct.
$1 \leq c_i \leq 10^5$
All buildings are reachable from one another.

Output Format

Output a single line containing a single integer denoting the minimum number of monies that need to be spent to deliver all k bath soaps.

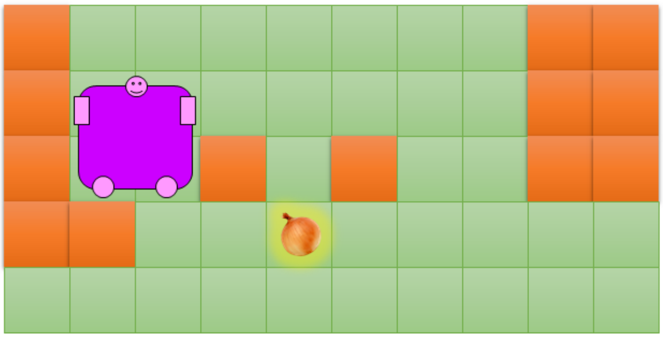
Sample Input	Sample Output
11 12 3 4 1 2 10000 2 3 10000 1 3 20000 3 4 17000 2 5 23000 5 6 10000 6 2 2000 6 7 6000 7 8 6000 8 9 100 9 10 100000 11 10 100000 4 8 11 9 3 1 5 7 6 10	181000

Problem F
Yumamma II
Time Limit: 3 seconds

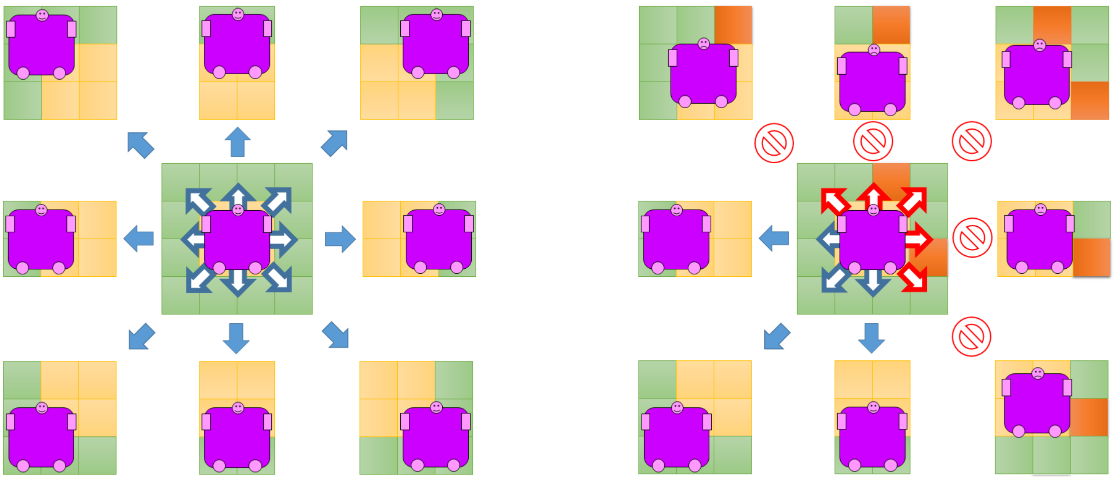
Throughout her whole life, Yumamma wanted to be in a programming problem involving a maze. However, most characters in mazes need to fit in a 1×1 cell. Thus, she never gets chosen to be a character. This is because Yumamma's so fat that she must occupy a 2×2 square in a grid.

Today is Yumamma's lucky day. We have created a problem just for her! Here it is:

Yumamma's so ugly, she once made an onion cry. Now, she wants to apologize to this onion and go to its house, which is a 1×1 cell. The map to go to the onion's house is representable by an $r \times c$ grid.



Yumamma's so fat, she can't carry herself to walk anymore. She has to slide. She can slide one step towards any of the four cardinal directions or one step towards any of the four diagonal directions as long as she will not hit any obstacles along the way.



Yumamma's so old that she should not waste time pointlessly wandering around the map because she feels her mortality is creeping up on her already.

What's the least number of steps Yumamma must make so that one-fourth of her entire body is in the same cell as the house of the onion?

Input Format

The first line of input contains t, the number of test cases.

The first line of each test case contains two space-separated integers r and c denoting the number of rows and columns of the grid, respectively. The ith following line contains a string of length c consisting of the characters `Y`, `#`, `O` and `.` (dot).

`Y` denotes Yumamma's initial location, `#` denotes an obstacle, `.` (dot) denotes free space, and `O` denotes the onion's house. There will be exactly four `Y`s in the grid which will appear in a 2×2 subgrid, and there will be exactly one `O` in the grid.

Constraints

$1 \leq t \leq 10^5$
$2 \leq r, c \leq 1000$
$rc > 4$
The sum of rc across all cases is $\leq 3 \cdot 10^6$

Output Format

For each test case, output a single line containing a single integer denoting the least number of steps needed for Yumamma to reach the onion's house. If it's impossible, output -1 instead.

Sample Input	Sample Output
3 5 10 #.......## #YY.....## #YY#.#..## ##..O..... .... 8 8 .... .YY.#... .YY..... ....#..... ....O .... 5 5 ....O#. ..YY. .#YY.	11 5 -1

Problem G

Win

Time Limit: 3 seconds

Little Angelo, the young helicopter, was taught at an early age that winning is everything and that success can only be measured by the medals you gain and contests you win.

In fact, to ingrain this principle into his head, his helicopter parents bought him three different types of alphabet blocks when he was a toddler – labeled "**W**", "**I**", and "**N**". It was only until Little Angelo became a teenage helicopter that he finally bought "**F**", "**U**" blocks to remind himself that in addition to wanting to "**WIN**", one must also have "**FUN**".

But when he was still a toddler, Little Angelo's parents would play a game with him every day. His parents would arrange the various "**W**", "**I**", "**N**" alphabet blocks they bought for their son into an $r \times c$ grid. Little Angelo then had to find sequences of three edge-adjacent blocks containing "**W**", "**I**", "**N**", in that order. Every time he finds such a set, he must remove the three blocks, add one point to his score, and continue finding more with the remaining blocks. He cannot rearrange the remaining blocks at any point.

To motivate their child, the helicopter parents hover over their son the entire time. In addition, if their son finishes with a non-optimal score, they punish him by disallowing him to play during the weekends – the only time Little Angelo is allowed to play in the first place.

Given a grid of blocks, what is the maximum possible score one can achieve in the game? Also, give a sequence of moves achieving that score.

Input Format

The first line of input contains t, the number of test cases. The following lines describe the test cases.

The first line of each test case contains two space-separated integers r and c. The ith of the next r lines contains a string of length c consisting of the letters `W`, `I` or `N`, denoting the ith row of the grid.

Constraints

$1 \leq t \leq 5$
$1 \leq r, c \leq 60$

Output Format

For each test case, output a single line containing a single integer s, denoting the maximum possible score one can achieve in the game. Then, output s lines indicating a sequence of moves to achieve a score of s.

Each such line must contain six space-separated integers $i_1, j_1, i_2, j_2, i_3, j_3$. (i_1, j_1), (i_2, j_2) and (i_3, j_3) are the coordinates of blocks that spell out "**WIN**", i.e.,

- Cells $(i_1, j_1), (i_2, j_2), (i_3, j_3)$ contain "**W**", "**I**", "**N**", in that order.

- Cells (i_1, j_1) and (i_2, j_2) are edge-adjacent.
- Cells (i_2, j_2) and (i_3, j_3) are edge-adjacent.

The coordinates (i, j) refer to the block at the ith row (from the top) and the jth column (from the left). The sequence of moves must be valid; in particular, a block must not be removed twice.

There may be multiple valid sequences of moves; any one will be accepted.

Sample Input

```
1
4 6
IIWWII
WINNWN
IWINII
NWWINW
```

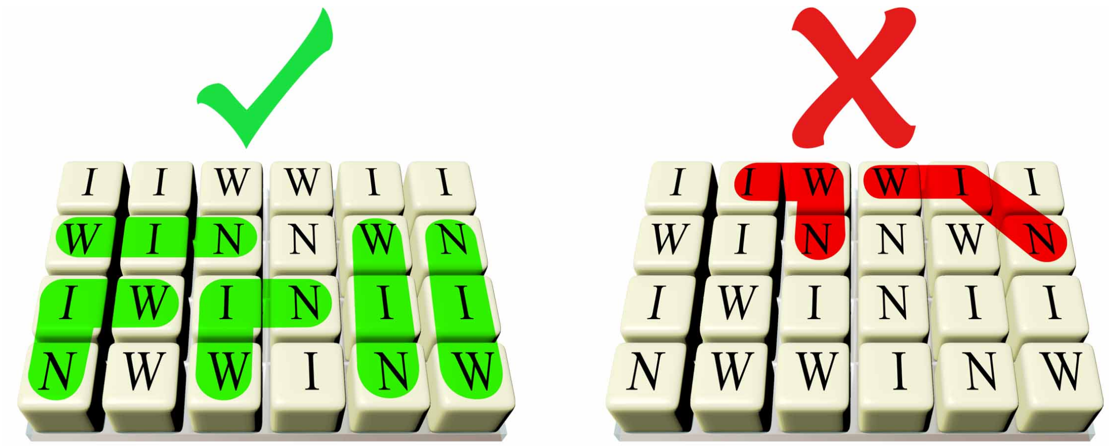
Sample Output

```
5
4 3 3 3 3 4
2 1 2 2 2 3
2 5 3 5 4 5
4 6 3 6 2 6
3 2 3 1 4 1
```

Explanation

The answer for the sample input is 5. One way to achieve this score is illustrated by the first image below.

On the other hand, each set of three blocks highlighted in the second image below is not allowed. The first one is invalid since "**WIN**" is not in the right order, while the second one is invalid since "**I**" and "**N**" are not edge-adjacent.



Problem H
Aqua Man's Aqua Room
Time Limit: 2 seconds

Aqua Man has an aqua room, an empty room in which water, and only water, is intended to be kept. It is not to be confused with his sibling Aqua Mian's aquarium, which is intended to be the habitat of fish and other sea creatures.

When viewed from the top, the aqua room has dimensions r meters $\times$ c meters. From this position, this aqua room can be viewed as a 2D grid consisting of $r \times c$ cells of size 1×1.

The rows are indexed from 1 to r from top to bottom and the columns are indexed from 1 to c from left to right. We denote by cell (i, j) the cell which is in the ith row and jth column.

Each cell is flat and thus has a constant height. From now on, we shall measure heights relative to the height of cell $(1, 1)$, which we will set to have height $h_{1,1} = 0$. We write $h_{i,j}$ for the height of cell (i, j). This means that cell (i, j) is $h_{i,j}$ meters higher than cell $(1, 1)$.

The ceiling of the room is also flat. It is h meters above cell $(1, 1)$ and thus, the room is said to have height h.

Another property of this room is that for any cell (i, j) in the $r \times c$ grid, the height of cells (k, j) for any $k > i$ is at least $h_{i,j}$ and the height of cells (i, k) for any $k > j$ is at least $h_{i,j}$.

Cell $(1, 1)$ has a hole in the ceiling which for some reason drips beer continuously at a rate of 1 cubic meter per millennium. Most people would be happy about this fact. Unfortunately, Aqua Man's religion states that all of its followers are not allowed to touch beer at any point in their lives. In fact, doing so – accidentally or intentionally – is a mortal sin.

If the room starts empty and the hole is allowed to drip for exactly v millennia, how many cells in the room can Aqua Man step on such that he does not commit a mortal sin? Assume Aqua Man is still alive at this point and that the room is never changed in any other way.

You need to answer this question q times, each consisting of a different value of v.

Input Format

The first line of input contains four space-separated integers r, c, h, and q.

The next r lines contain c integers each. This gives the terrain's height information. Specifically, the jth number in the ith line contains $h_{i,j}$, the height of cell (i, j).

The next q lines describe the queries. Each query consists of a single line containing a single integer v.

Constraints

$1 \le r, c \le 500$
$1 \le h \le 10^6$
$0 \le h_{i,j} < h$
$h_{1,1} = 0$

$v \geq 1$

v cubic meters of beer will fit inside the room.

$1 \leq q \leq 10^5$

Output Format

For each query, output a single integer in a single line denoting the number of cells in the room Aqua Man can step on such that he does not commit a mortal sin.

Sample Input	Sample Output
6 5 10 2 0 1 1 3 4 2 2 3 4 6 2 3 3 5 6 2 3 6 7 8 4 5 7 8 9 5 6 7 8 9 4 5	27 23

Problem I

Rainbow Dash

Time Limit: 4 seconds

New roads have become available in the magical kingdom of Princess Celestia and the citizens of Equestria want to have rainbow-colored road markings instead of the usual boring white one. The kingdom has n different locations and it has a total of h highways, each of which starts from one of the locations and leads to another. Take note that it is possible that several highways may start and end at the same pair of locations.

Princess Celestia only requires r highways to be painted. She calls these the *royal* highways. She chose the set of royal highways such that if someone starts from any of the n locations, he or she will be able to reach all the other locations by exclusively passing through royal highways.

Rainbow-colored paint is expensive. Luckily, since Twilight Sparkle is a unicorn, her digestive tract can produce rainbow-colored paint. However, she currently has diarrhea and cannot control the paint coming out from her body. Thus, she spreads rainbow-colored paint wherever she walks!

Starting from any of the n locations, is there a path which Twilight Sparkle can follow such that she can paint all royal highways? Take note that Twilight Sparkle can optionally pass by non-royal highways in this path. However, Twilight Sparkle should not pass by the same road twice since the paint might still be wet.

If there is such a path satisfying the conditions above, output a sequence of highways which Twilight Sparkle must dash through. Otherwise, say that there is no such path.

Input Format

The first line of input contains t, the number of test cases. The following lines describe the test cases.

The first line of input contains three space-separated integers n, h, and r denoting the number of locations, highways and royal highways, respectively. The locations are numbered from 1 to n. The next h lines describe the highways. The ith line contains two space-separated integers a_i, b_i, indicating that the ith highway connects locations a_i and b_i. The first r highways listed are the royal highways.

Constraints

$1 \leq t \leq 5000$
$1 \leq n \leq 10^5$
$0 \leq r \leq h \leq 2 \cdot 10^5$
The sum of the hs is $\leq 6 \cdot 10^5$
$1 \leq a_i, b_i \leq n$

Output Format

For each test case, output a single line containing either the word `yes` or `no`. In addition, if you answered `yes`, output two more lines. The first line contains a single integer m denoting the number of highways in the path. The second line contains m integers $i_1, i_2, \ldots, i_m$ denoting the indices of the highways in the path, in the path order.

Sample Input	Sample Output
2 6 8 5 1 2 2 3 2 4 1 5 1 6 1 3 3 4 1 4 5 6 4 1 2 2 3 2 4 1 5 1 4 3 4	no yes 5 4 1 2 6 3

Problem J

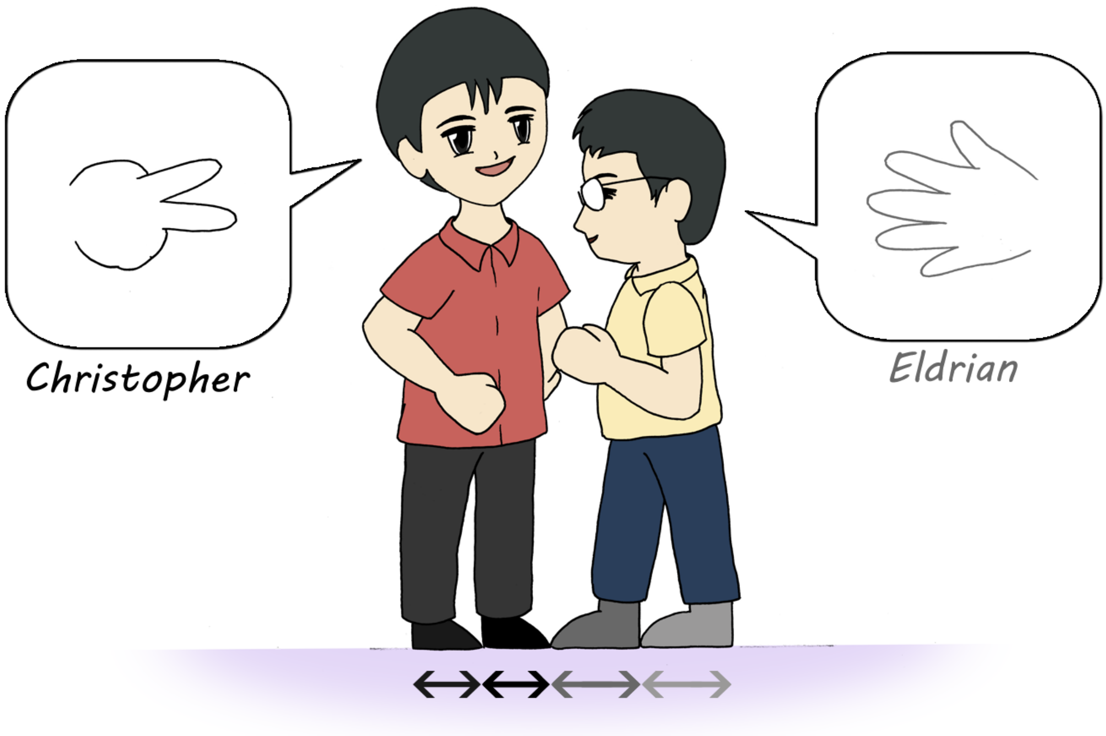
Bato Bato Split

Time Limit: 2 seconds

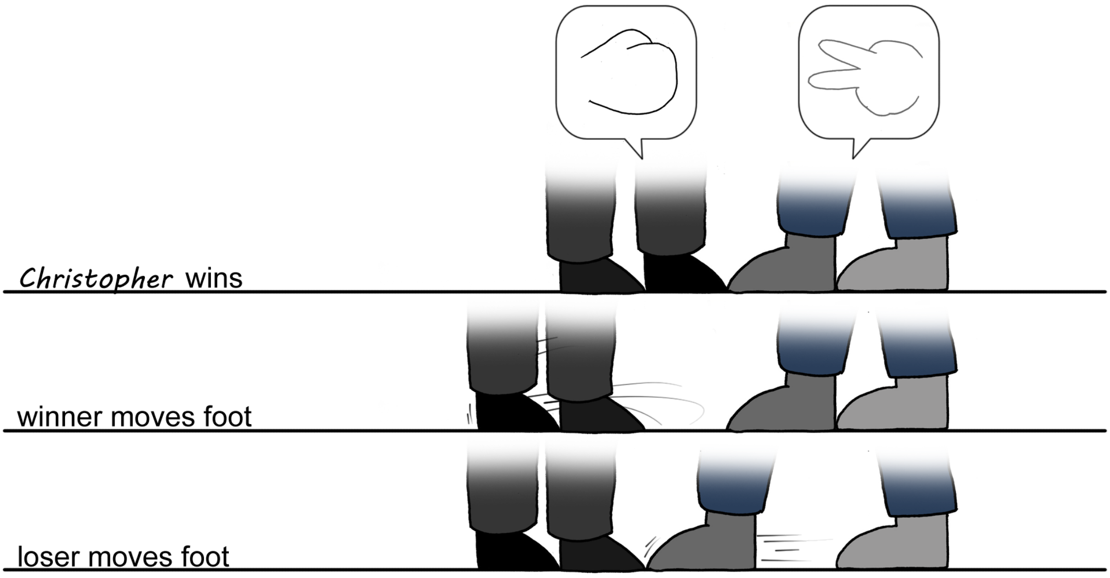
Bato Bato Pick, more commonly known as Rock Paper Scissors, is a game where two players simultaneously make a gesture representing one of the following: a rock, paper, or scissors.

The rules are as follows: paper beats rock, rock beats scissors and scissors beat paper. If both players make different gestures, then the player who made the gesture which beats the other wins that round and gets a point. If both players make the same gesture, that round is a draw and none of them gets a point.

Bato Bato Split is a variation of this game. In this game, players begin by facing each player and must make a line segment on the ground using their feet. In this position, one of their feet will be in front and the other on the back. The front tip of the front feet of both players must touch each other. This process is illustrated below.

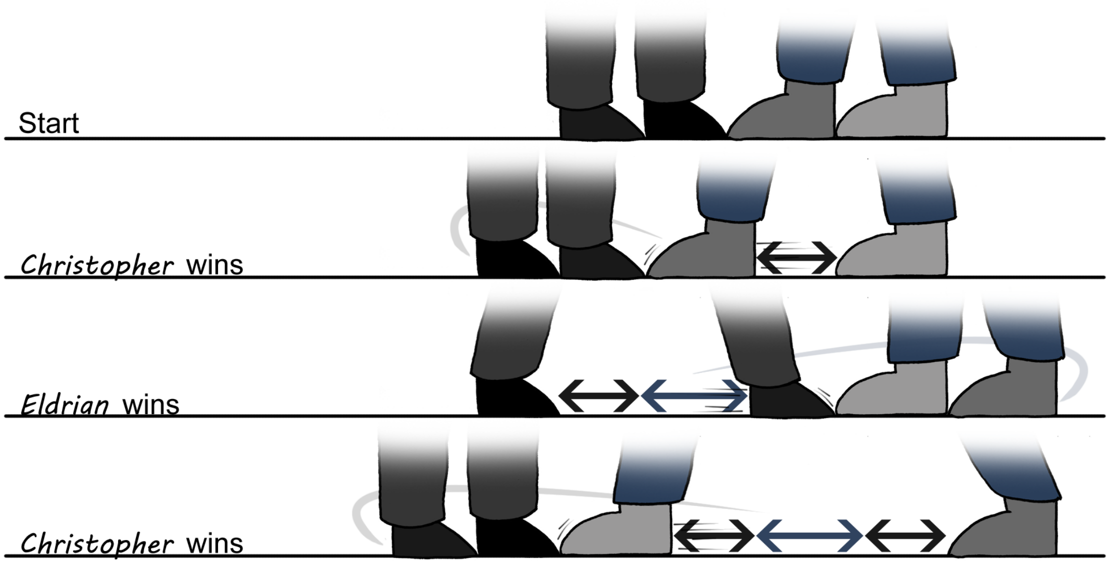


The first round begins. Both players do one round of Bato Bato Pick. Whoever wins the round puts their front feet directly behind their back feet. The loser must then reconnect his front foot to the new front foot of the winner. In other words, the front foot of the loser must slide forward to touch the winner's front foot, while the back foot of the loser must stay in place. This process is illustrated below.



Once this is done, a new round starts and the winner and loser do the actions described in the previous paragraph again. If the loser cannot do his task, then he loses the whole game.

Here is a sample of the feet movement for the first three rounds:



Best friends Christopher and Eldrian intend to play Bato Bato Split. They are very fond of this game because their bodies are quite flexible. In fact, they can separate their feet up to a distance of d units!

Given the length of the feet of each player, find a series of n moves for both players so that after these n bato bato pick rounds, the distance between the front tips of the two feet of a particular player is exactly d units. If this is impossible, say so.

Assume that the lengths of the feet of a single player are equal.

Input Format

The first line of input contains t, the number of test cases.

The each test case consists of a single line containing four integers n, d, f_C, f_E and a letter p, separated by single spaces. f_C denotes the length of each foot of Christopher. f_E denotes the length of each foot of Eldrian. p is either `C` or `E` and denotes the first letter of the player whose tips should be of distance d.

d, f_C, and f_E are measured in the same units.

Constraints

$1 \le t \le 2 \cdot 10^5$
$1 \le n \le 2 \cdot 10^5$
$1 \le d, f_C, f_E \le 10^9$
p is either `C` or `E`.
The output size is at most $4 \cdot 10^6$ bytes.

Output Format

For each test case:

- If it's impossible to find such sequences of moves, print a single line containing the string `IMPOSSIBLE`.
- Otherwise, output two lines, each containing a string of length n consisting of the letters `R`, `P` and `S` describing the sequence of moves of Christopher and Eldrian, respectively. `R` represents *rock*, `P` represents *paper*, and `S` represents *scissors*.

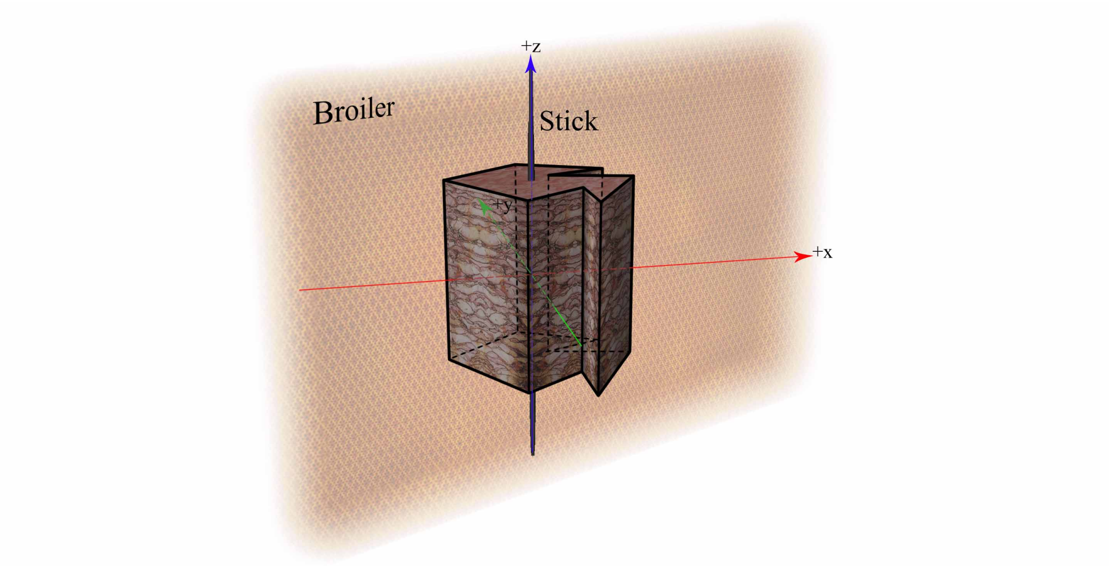
Sample Input	Sample Output
2 5 2 4 3 C 5 8 1 2 E	IMPOSSIBLE PRSSR RPRPR

Problem K
Kebab
Time Limit: 4 seconds

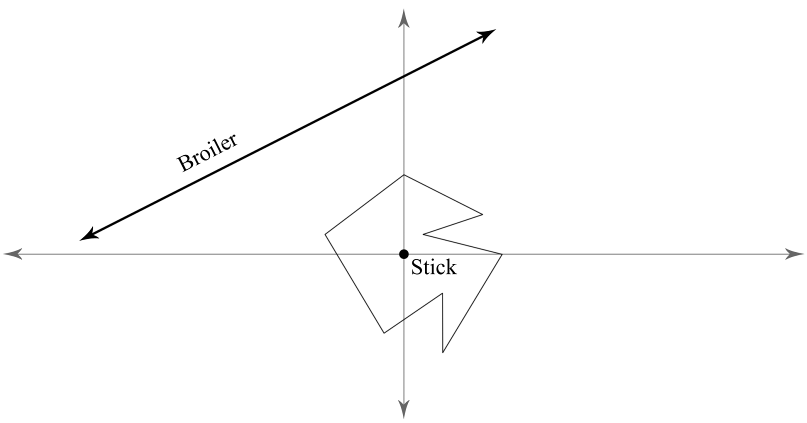
A kebab machine is a machine with two parts. The first part is a **stick** in which you are supposed to place the **meat**. The second part is a vertical **broiler** to heat it up.

We are cooking a piece of meat with a kebab machine, and we want to compute the expected amount of heat that it receives. Let's formalize the problem.

We assume that the meat's shape is a *right prism* with polygonal bases. We position it so that the bases are horizontal, i.e., parallel to the xy-plane. Also, the kebab machine is positioned so that the stick and broiler are vertical, and the stick coincides with the z-axis. Since the meat is skewered, we can assume that the stick passes through the *interior* of the prism.



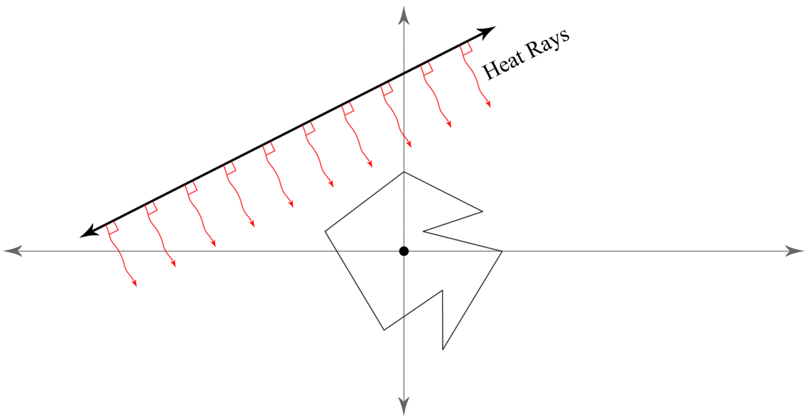
Note that we're modelling the objects in 3D space. However, due to the way we oriented the objects, notice that we can simplify our model by just looking at our setup from above:



Thus, we can model everything in the 2D plane: the meat as a simple polygon containing the origin, the stick as the origin, and the broiler as a line L determined by some equation

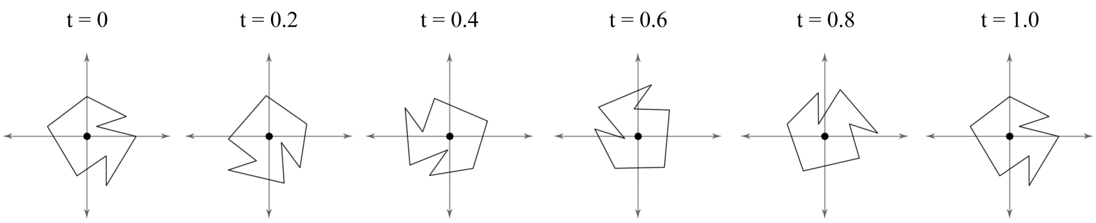
$ax + by = c$. (Who knew describing a piece of meat would be so tedious? But we're not done yet. Remember that our goal is to compute the amount of heat that the meat receives.)

Heat rays emanate from the broiler in a direction perpendicular to L and towards the direction of the origin:



More formally, heat rays continuously emanate from the broiler towards either the direction $\langle a, b \rangle$ or $\langle -a, -b \rangle$, towards whichever side of L the origin is located. The broiler emits heat rays *uniformly* from its surface, so there's a heat ray emanating from every point of the broiler. (The image above only shows a few heat rays for clarity.)

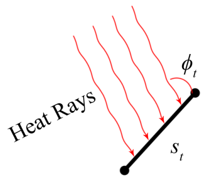
Now, the meat is slowly rotated so that the sides can be cooked by the broiler. Specifically, the meat continuously rotates around the origin clockwise at a rate of one full rotation per minute. Let t denote the time elapsed since the beginning of the cooking process, in minutes. Thus, the cooking starts at $t = 0$, and at $t = 1$, the meat has made one full rotation. We will assume that L is far enough away that the polygon will never intersect L at any point in time.



We are now ready to compute the amount of heat that the meat receives.

Let S be some side of the meat. We define the **amount of heat** that S experiences at time t as $s_t \sin \phi_t$, where:

- s_t is the length of the portion of S that is being hit by heat rays at time t, and
- ϕ_t is the angle made by the heat rays with S at time t.



The meat is opaque, and we assume that heat rays cannot pass through opaque objects. Thus, s_t could be less than the length of S, for example when S is partially obscured by other parts of the meat. Note that if S is totally obscured, or if S is facing away from L, then $s_t = 0$.

The **total amount of heat** the meat gets at time t is just the sum of the amounts of heat experienced by all its sides at time t. We denote this as $H(t)$.

Assume that t is chosen from the interval $[0,1]$ uniformly at random. What is the expected value of $H(t)$?

Input Format

The first line of input contains a single integer k denoting the number of test cases. The description of k test cases follow.

The first line of each test case contains four space-separated integers n, a, b and c. n describes the number of vertices of the polygon representing the meat, and a, b and c describe L as the line determined by the equation $ax + by = c$.

The next n lines describe the coordinates of the polygon at time $t = 0$, in counterclockwise order. Specifically, the ith line contains two space-separated integers x_i and y_i denoting that the ith vertex of the polygon is located at (x_i, y_i) at time $t = 0$.

Constraints

$1 \le k \le 3 \cdot 10^4$
$3 \le n \le 10^5$
The sum of the ns in a single file is $\le 3 \cdot 10^5$
$|x_i|, |y_i| \le 10^8$
The polygon is simple.
The polygon strictly contains the origin.
$|a|, |b|, |c| \le 10^9$
$(a,b) \ne (0,0)$.

$c \neq 0$.

The line L will never intersect the polygon at any point in time.

Output Format

For each test case, output a single line containing a single real number denoting the expected value of $H(t)$. Your answer will be considered correct if it is within an absolute or relative error of 10^{-6}.

Sample Input

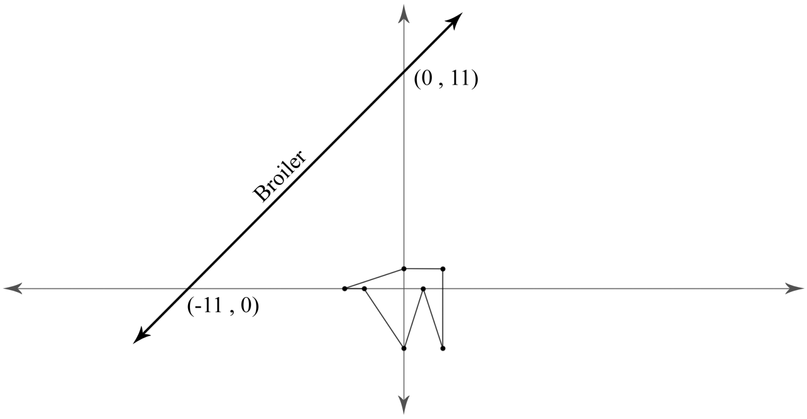
```
1
7 -1 1 11
0 -3
1 0
2 -3
2 1
0 1
-3 0
-2 0
```

Sample Output

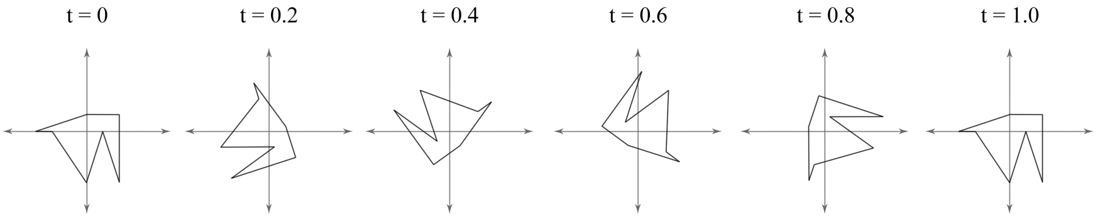
```
4.9035380
```

Explanation

The setup in the sample input looks like this:



Here are some snapshots of the meat at various points in time, as it rotates clockwise.



Problem L

Frickin' Heck

Time Limit: 2 seconds

Christian keeps n jars of sweets all lined up in a row. The ith jar from the left contains a_i sweets. Despite his passion for eating sweets, he wants to improve himself by eating healthier.

Christian's mother is a very strict religious woman. She has banned everyone in their house from saying the two bad words. Whenever Christian says any of these words, the mother forces him to throw away sweets from his jars.

The first of the two bad words is *heck*. This is quite unfortunate for Christian because he occasionally jokingly tells his friends to "go to heck". Whenever Christian says *heck*, his mother would force him to choose three adjacent jars, all of which should be non-empty, and then throw away exactly one sweet away from each of them.

The second of the two bad words is *frick*. Christian commonly says this whenever he does something cool. He tells himself that he's "lit as frick". Whenever Christian says *frick*, his mother would force him to choose one jar with at least two sweets, and then tell him to take exactly two sweets away from the jar and throw them away.

In either case, there should be enough sweets in the jars for the move to be possible, otherwise, no sweets are thrown away at all.

Since Christian wants to stay away from sweets anyway, he decides to turn the punishments into a game. He wants to know if he can manage to empty all n jars just by saying bad words.

Is it possible to do that? If so, what is the fewest number of bad words he must say?

Input Format

The first line of input contains t, the number of test cases. The following lines describe the test cases.

The first line of each test case contains a single integer n. The second line contains n space-separated integers $a_1, a_2, \ldots, a_n$.

Constraints

$1 \le t \le 10^5$
$1 \le n \le 2 \cdot 10^5$
$0 \le a_i \le 10^8$
The sum of the ns is $\le 8 \cdot 10^5$

Output Format

For each test case, output a line containing either a single integer denoting the fewest number of bad words he must say, or the word `no` if it is impossible to empty all jars.

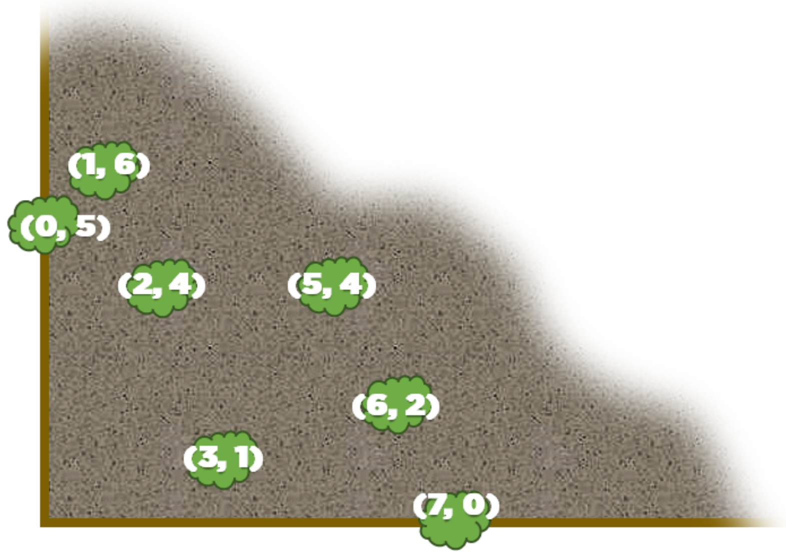
Sample Input	Sample Output
2 4 1 4 2 1 5 2 0 1 0 2	3 no

Problem M

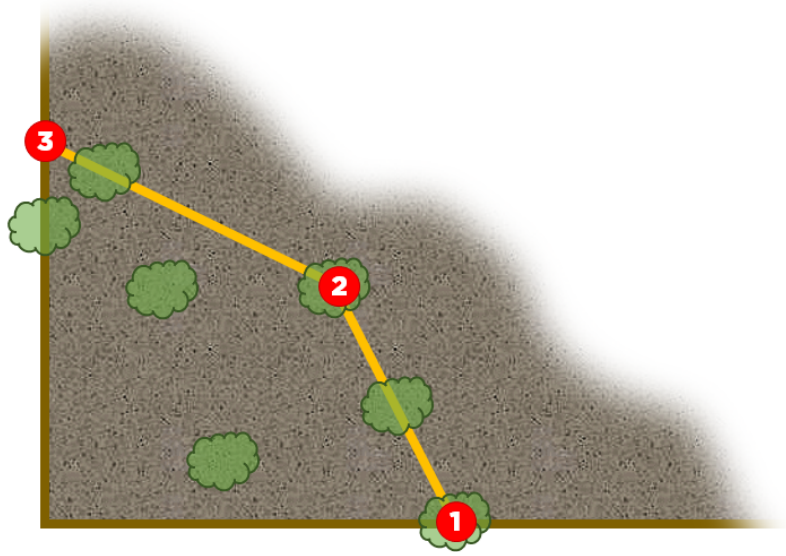
Danielrad Cliff

Time Limit: 5 seconds

Danielrad Cliff is interesting in that it has a 90 degree angle. From a bird's eye view, we can model the cliff as the *first quadrant* of the xy-plane. Somewhat close to the edge of this cliff, there grew n endangered plants. The local government decided to erect a barrier to prevent unsuspecting animals and people from harming the plants.



The barrier shall involve three posts – and we index them from 1 to 3. Post 1 shall lie on the x-axis, Post 3 shall lie on the y-axis, and Post 2 can be anywhere.



Once the locations of the three posts are decided, consecutively-indexed posts will each be connected by a straight electric fence. Fencing one meter costs 1 thousand pesos. If we choose the positions of the three posts wisely, we should be able to end up with fewer expenses involving the electric fence.

After installing the two electric fences, all n plants should be inside the region R enclosed by the fence and the cliff. Plants that are directly on the fences are considered to be inside R; assume that the construction company can find a way to make sure that the plants in the way of the construction of the fences are inside R.

What is the least amount of money, in thousands of pesos, that needs to be spent on the electric fence if we want to enclose all n plants using the method described above?

Input Format

The first line of input contains t, the number of test cases.

The first line of each test case contains a single integer n denoting the number of plants. The ith of the n following lines contains two space-separated integers x_i and y_i. (x_i, y_i) is the location of the ith plant. We assume that Danielrad Cliff is represented as the first quadrant in the xy-plane.

Constraints

$1 \le t \le 150$
$1 \le n \le 20$
$0 \le x_i, y_i \le 1000$
The locations are distinct.

Output Format

For each test case, output a single line containing a single real number denoting the answer for that test case. The answer must be correct up to an absolute or relative error of 10^{-5}.

Sample Input	Sample Output
2 7 0 5 7 0 1 6 2 4 3 1 5 4 6 2 2 0 1 1 0	10.06230590 1.414213563