Problem Statements

in

Insomnia'13

<15th March 2013>

1. Visit to biolab

Time Limit = 3 sec
Number of Attempts = 175
Number of Correct Solutions = 33

Digo is a science student. Recently he visited his school Biology Lab. There he was told about a newly discovered bacteria. He found that the bacteria is circular with a radius epsilon(say 10^-15 m). Now, he is curious about colonial growth pattern of that bacteria.
So, he placed some bacteria at various spots in a rectangular tray which is overlaid with a coordinate system. Surprisingly he observed that if bacteria are present at (x1,y1) and (x2,y2) coordinates then after sometime there will be a new bacteria at ((x1+x2)/2,(y1+y2)/2) coordinate also. Bacteria can overlap and their center of coordinates will have integer values only. Digo led the process for quite a long period of time until a stable(non changing) pattern is obtained. Digo also feels that the final number of bacteria and area covered by bacteria must have a relation. To verify this he assigned you a portion of the task. He wants you to write a program to calculate the TWICE the area covered by the colony.

INPUT FORMAT:
First line has an integer T which shows the number of test cases.
First line of each test case has an integer N which shows the number of bacteria in that test case.
Next N lines have two integers Xi Yi, with a space in between, which show the X and Y coordinates of that bacterium.

CONSTRAINTS:
1 <= T <= 1000
1 <= N <= 1000
0 <= Xi <= 1000000
0 <= Yi <= 1000000

OUTPUT FORMAT:
For each test case, print a line having TWICE the area that will be covered by bacteria colony after a long time.

SAMPLE:
Input:
3
4
0 0
3 3
6 6
10 10

1
10 240
4
10 49
15 49
3 12
37 8
Output:
0 //Explanation: All Bacteria will be in a single line
0 //Explanation: Only one bacteria is there
1491

2. Running late for office

Time Limit = 1 sec
Number of Attempts = 21
Number of Correct Solutions = 4

Digo got an opportunity to be an intern at code village and was residing at the guest house provided to him. He was new to the village and didn't know how to reach his office. He was given the instructions of where the office is located relative to the guest house's position. As you all know, code village is famous for the criss cross pattern of roads. To put it mathematically, assume that the city of code village is a square grid where each grid point is a junction and the lines connecting the grid points are the roads. Assume that the guest house is located at (0,0) and his office is located at (X,Y). You need to find the number of ways in which Digo can reach the office. Note that Digo doesn't want to reach the office late so he will choose the shortest path. To make matters worse, he received information that there is lot of congestion at various junctions so he would not want to go through these junctions or else he will get late for his office.

Input :

First line of the input contains the number of test cases T. Then T cases follow each having the following format The first line of each test case contains three space separated non-negative integers X, Y and N respectively. Then the next N lines, each containing a pair of non-negative integers, represent the co-ordinates of the junctions (x[i],y[i]) where there is congestion.

Output :

Print the number of ways Digo can reach his office using the shortest path(of distance X+Y) and also avoiding the congested junctions. As the answer could be very large, print it modulo (10^8 + 2013).

Constraints :

1 <= T <= 10
0 <= X,Y <= 10^5
0 <= N <= 1000
0 <= x[i] , y[i] <= 10^5 for all 1 <= i <= N
It is assured that all the congested junctions are distinct and do not overlap with (0,0) or (X,Y).

Sample Input:

```
3
3 3 1
2 2
3 3 2
1 1
2 2
1 1 2
0 1
1 0
```

Sample Output:

```
8
```

4
0

3. Let's play a game

Time Limit = 2 sec
Number of Attempts = 79
Number of Correct Solutions = 8

It is a lesser known fact that Digo is a philander. Did we tell you about his two girlfriends Diggi and Duggu? Digo keeps asking puzzles to them. They are bored of answering so many questions which he asks. So, they decided to play a game. As they both love math, their game is also related with math and matrices. The game is as follows: A matrix of N x M positive integers is formed. During his turn, the player chooses a row, which has at least one non-zero cell. From that chosen row, the player can choose the first non-zero cell from the front or the back. Note that here we treat the row as a list of numbers. Let the chosen cell has the value X. Now, the player can subtract any number from 1 to X from that cell.

/*There will be only one end(cell) to choose when only one non zero cell is present in the chosen row.*/

Diggi always takes the first turn in any game and afterwards both of them take alternate turns. The player who can't make a further move loses the game.
Digo heard their discussion about the game. He knows that both Diggi and Duggu are super intelligent and will always make the optimal move. Suddenly a thought arose in his mind. He thought that for any given matrix it can be predicted that who is going to win the game. And what!! He got the method to predict this. Ahh.... But he is quite cunning and does not reveal the method. Instead he challenges you to solve the problem.

INPUT FORMAT:
T (T is number of test cases.)
N1 M1 (N1 is number of rows and M1 is number of columns, for the matrix given in first case.)
Next N1 lines have M1 positive integers.
N2 M2
Next N2 lines have M2 positive integers.
....
....
NT MT
Next NT lines have MT positive integers.

CONSTRAINTS:
T <= 30
1 <= Ni, Mi <= 20
0 < Initial value at each cell < 10

OUTPUT FORMAT:
For each test case print answer in a new line.
Print "Diggi", without quotes, if Diggi is going to win the game else print "Duggu", without

quotes when Duggu is going to win the game.

SAMPLE:
Input:
2
1 4
2 4 1 3
2 3
4 1 6
7 1 5
Output:
Diggi
Duggu

4. Let's destroy them

Time Limit = 1 sec
Number of Attempts = 57
Number of Correct Solutions = 4

Digo got a new missile system. There are N cities present in a row in the enemy country. Cities are indexed from 1 to N from left to right. When a missile is launched on a city it not only damages that city, but also damages the neighbouring cities. The missle also has a damaging range D. That means when a bomb is dropped on city with index 'i' then all the cities with index 'j' such that (|i-j|<D) are damaged by (D-|i-j|). Simply to say city with index 'i' is damaged by D, its immediate neighbours are damaged by D-1 and their immediate neighbours are damaged by D-2 so on upto damage 1. Missile Launching System(MLS) is built with the state-of-art AI technology. Digo has Q operations to process on the MLS. In one operation he can either order the MLS to drop a bomb with damage range D on city 'i' or he can ask the MLS to report the total damage done till now to a range of cities from index 'i' to index 'j'.
You have been recruited into the secret services and your first project is to handle the requests which Digo makes. You want to impress your seniors by designing an efficient system which can handle huge number of requests so that you outrun your enemies.

Input:
The first line contains an integer N representing the number of cities. The next line contains an integer Q which is the number of requests made by Digo.
Q lines follow each representing a request. Each of which can be one of the following two types.
Each line has three space separated integers x, y, z
'x' represents the type of query. If x is equal to 0 then you have to process type 0 query, which means a bomb is dropped on city with index 'y' with a damage range 'z'.
If x is equal to 1 then you have to process type 1 query, which means you have to print the sum of damages to the cities with index 'i' such that (y <= i <= z)

Output:
For every request of type 1 print an integer on a new line. As the result can be very large so output it modulo 10^8 + 2013.

Sample Input:
5
3
0 3 2
0 1 2
1 1 5

Sample Output :
7

Explaination :
First request is of type 0 so a bomb is dropped on city 3 with damage 2, so cities 2 and 4 are also damaged by 1 each.
After the first request damages done to the cities are:
Cities : 1 2 3 4 5
Damage : 0 1 2 1 0
Second request is of type 0 so a bomb is dropped on city 1 with damage 2, so city 2 is also damaged by 1.
After the second request damages done to the cities are:
Cities : 1 2 3 4 5
Damage : 2 2 2 1 0
Third request is of type 1 which asks for the sum of damages done to the cities with indices from 1 to 5. Sum of damages done to the cities = 2 + 2 + 2 + 1 + 0 = 7.
So the result is 7.

Constraints:
1 <= N <= 10^6
1 <= D <= N
1 <= Q <= 10^5
If x = 0, 1 <= y <= N and 1 <= z <= N
If x = 1, 1 <= y <= z <= N
0 <= result < 100002013

5. Lab

Time Limit = 3 sec
Number of Attempts = 179
Number of Correct Solutions = 19

The lab of codevillage has N systems connected to each other such that there is exactly one path between any 2 systems. Data can be transferred in both directions and it can also be transferred via intermediate systems. The details of connections and the lengths of wires between the systems will be provided to you. You have to answer Q queries, each of the form x y z (all 3 distnct). Each query is such that x and y are trying to communicate and z has failed. Will the communication between x and y suffer? You have to answer "YES" or "NO" (quotes for clarity only). Note that queries depict independent scenarios i.e. in no way do they affect each other. Moreover if the communication suffers then the lab incharge Digo will repair the faulty system and then will replace all the wires along the communication path between x and y. In such a case you also have to tell the length of the wire required to reconnect x and y.

INPUT:
First line consists two space separated integers N and Q.
Each of the next N-1 lines are of the form
u v len
meaning that there is a direct bidirectional connection between the systems u and v, and the cable connecting them is of length len. This is followed by Q lines, each of the form:
x y z
All three of which are distinct.

OUTPUT:
For each query output the result on separate lines. [See examples for more clearity]

CONSTRAINTS:
1 <= N <= 10000
1 <= Q <= 500000
u != v 1 <= u,v <= N
1 <= len <= 10^9
No pair of (u,v) will be repeated In the queries x, y and z are distinct and 1 <= x,y,z <= N

SAMPLE INPUT
5 4
1 2 1
1 3 1
2 4 1
3 5 2
1 2 3

2 3 1
4 3 5
4 5 3

SAMPLE OUTPUT
NO
YES
2
NO
YES
5

Explaination for first sample i/o
For first query, 1 and 2 try to communicate and 3 goes down, but communication between 1 and 2 will not suffer. For 2nd query, 2 and 3 try to communicate and 1 goes down. So communication will suffer. Moreover when communication suffers, then we are supposed to calculate the length of wire required to replace wires along the path from 2 to 3. In this case the required length is 1+1=2.

6. Game

Time Limit = 2 sec
Number of Attempts = 11
Number of Correct Solutions = 4

Digo works at codevillage, but since he likes playing games, he wants the name to be changed to gamevillage. For this he has to defeat the president of codevillage in a game. The game they planned to play is a 2-player board game, where players get alternate turns. Digo will play the first chance. The game is such that the board contains N blocks numbered 1 to N, and has exactly one token which will be moved by both players. Initially the token is at position K.

Since everyone at codevillage loves maths, they decided to have some mathematics in the game. In a chance, if the token is at block 'a', player may move it to a block 'b' iff (b>a and b is divisible by a). Also to make things interesting, for a move to block 'b', a player would have to invest a sum of ((a+b)%M) rupees, where M would be pre-known. The player to make a move to block N is declared winner. If this doesn't happens i.e. if the token is not at N and there is no move available, the match is declared a draw.

Digo realised that it may not always be possible for him to win the game. So he bribed the president. Now president will make the moves as told by Digo. Now of course Digo would have to pay for the moves of both the players. Given values of N and M, your task is to find out whether it is possible for Digo to plan the moves so that he wins. Also if he wins find the minimum money for moves of both players which he would have to invest in the game. [See examples for more clarity]

INPUT:
First line consists of three integers n, k and m.
N=no. of blocks on the board

1<=N<=200000000000000 (i.e. 2*10^14)
1<=K<N
1<=M<=1000000007

OUTPUT:
Output "YES" (quotes for clarity) if it is possible for Digo to plan the moves so that he wins, other wise output "NO". Also in case of a yes, print the minimum money which Digo would have to spend.

SAMPLE INPUT
6 1 100

SAMPLE OUTPUT
YES
7

SAMPLE INPUT
6 2 100

SAMPLE OUTPUT
YES
8

7. Find The Largest Rectangle

Time Limit = 1 sec
Number of Attempts = 8
Number of Correct Solutions = 4

Shagun is a rich landlord from a village called Codevillage. He owns a large farm of land in the shape of a rectangle of dimensions l and w. There are many trees on his farm at different points(integer coordinates). Shagun loves trees so he won't allow tress to be cut down for any purpose. He wants to build a new house on a part of this farm. He wants a rectangular house and also doesnot want any trees to be cut. Now Shagun is very busy as he has to arrange construction material and labour for the work to start so he hires you to do this job. You have to calculate the area of the largest rectangular plot in the field which does not contain a tree where he can make his house. The boundary of his house may pass through a point where there is a tree but it must not contain the tree inside the choosen plot.

INPUT :
The first line of the input specifies the number of test cases. For each test case, the first line provides the length l and width w of the area in meters (0 For each test case you have to output the area of the largest rectangle which does not contain any tree within its boundary though trees can be present on the perimeter.

Constraints :

Number of test cases<=10
0 < length and width <= 10000
0<=Number of trees <=1000

Sample Input
2
2 2
2
0 0
1 1
3 3
3
0 0
1 1
3 3

Output
2
6

8. Digo wants burgers

Time Limit = 1 sec
Number of Attempts = 62
Number of Correct Solutions = 21

During his internship at codevillage, Digo got addicted to the burgers at McD. He recently got his stipend and wants to spend it on eating lots and lots of burgers. He has lot of money in his wallet. To be precise, he has D types of denominations. The ith denomination is of value V[i] and he has K[i] such coins i.e., he has K[1] coins of value V[1], K[2] coins of value V[2] and so on upto D. As Digo is crazy about math he wonders in how many ways can he tender a change for the burger he wants to eat using the coins in his wallet. You have to find the number of ways in which it is possible to achieve this for many queries. As the answer might be very large, output it modulo 10^8 + 2013.

Input Specification:
The first line contains a positive integer D which indicates the number of denominations available with Digo The next D lines follow each containing two integers V[i], K[i] indicating that the value of ith coin is V[i] and number of such coins is K[i]. Note that the information about the ith coin is given in the (i+1)th line.
Then on a new line a positive integer Q is given which denotes the number of queries to follow.
Then on the following Q lines a single postitive integer N is given which indicates the cost of the burger.

Output Specification:
Output Q lines, each of them containing a single non negative integer which indicates the number of ways in which Digo can buy the burger using the coins available with him. As the answer can be large, output it modulo 100002013.

Constraints:
1 <= D,V[i],K[i],N,Q <= 2000
0 <= answer < 10^8 + 2013

Sample Input:
2
2 3
3 2
2
2
6

Sample Output:
1
2

Explanation for sample input:
Using 3 coins of value 2 and 2 coins of value 3 we can tender a change for a burger of cost 2 in only one way whereas a burger of cost 6 can be bought in 2 ways.
2 = 2
6 = 2 + 2 + 2
6 = 3 + 3

9. Count them if you can

Time Limit = 1 sec
Number of Attempts = 56
Number of Correct Solutions = 7

As you all know Digo is not just a programmer, he is also a microbiologist. Recently he created a new virus which replicates very quickly. To be precise, if we have a virus at a particular point (x,y) then in one minute it dies and just before dying it gives birth to four other viruses which will be born at (x,y+1), (x,y-1), (x+1,y), and (x-1,y). Note that there maybe more than one virus at any point and they behave independent of each other.
Let us consider an infinite cartesian plane. Initially at time t = 0 minutes, Digo placed a virus at (0,0) and wonders how they might grow in number. Your task is very simple. You have to answer Q queries. Each query is represented as "x y z" (quotes for clarity) for which you have to find how many viruses infest the point (x,y) after z minutes.
As the answer can be huge so output it modulo 10^8 + 2013.

Input:
On the first line there will be an integer Q, representing the number of queries.
Q lines follow each having three integers x, y, z where (x,y) is the point under consideration and x represents the time elapsed from the start.

Output:
Output Q numbers each of them answering the queries given in the input. The ith number should answer the ith query.

Constraints:
1 <= Q <= 10^5
|x|,|y| <= 10^6
1 <= z <= 2*(10^6)
0 <= answer < 100002013

Sample Input:
3
0 0 0
0 0 1
0 0 2

Sample Output:
1
0
4

10. Catch me if you can

Time Limit = 2 sec
Number of Attempts = 10
Number of Correct Solutions = 4

Digo was the co-ordinator of Insomnia last year. He asked help for publicizing Insomnia by giving problems to participants for solving his financial problems during the contest, but the things are never as straight as they seem to be. He faced a bigger problem, the modey was getting stolen from the vault. But Digo is a smart boy, he got a new vault.

Again the things are not as straight as they seem!!!. The new vault turned out to be magical. In the vault, there are 100 slots in which dollar bundles can be put safely so that they may be used for publicizng the event. Now the bundles can be taken out from any slot and put to any other slot after some restrictions. Now the magic begins here:

1) A bundle can be moved from slot positon i to slot(i+2) or slot(i-2). For this move to happen the destination slot should be empty. And while performing this move, if there is a bundle in the slot between the destination and the source slots, the bundle would cease to exist after the move is completed.
2) A bundle can be moved from slot positon i to slot(i+2) or slot(i-2). For this move to happen the destination slot should be empty. And while performing this move, if the slot between the destination and the source slots is empty, the new bundle would appear in that slot after the move is completed.
3) A bundle can be moved from slot positon i to slot(i+3) or slot(i-3). For this move to happen the destination slot should be empty.

Now since Digo's finance team was not as good, it could manage at most 50 dollar bundles. Digo initially puts them among the first 50 occuring slots in some sequence. Now daily Digo goes to his vault and makes some changes in the pattern according to the restrictions given above. Digo moves only one bundle at a time. Many days passed this way. Suddenly Digo realizes he forgot to keep track of theft. Now he goes to his vault and opens to find that luckily only the slots among the first 50 contains the bundles so he will have to do less calculations (or he thinks so!!).

Now since Digo is busy in preparation of ACM-ICPC world finals, u have got to help him. U will be given an initial arrangement and some final arrangements, tell digo the possibility of theft for each of them. A theft is sure if the final arrangement cant be reached from the initial arrangement, otherwise not sure.

Input:

First line contains initial arrangement. `*` means there is bundle in the corresponding slot. `.` means the slot is empty. The given pattern will correspond to slots from left and u should assume other slots to be empty. pattern `*..*` means slot1 and slot4 have bundle and all other 98 slots are empty.

The next line contains `q`, number of final arrangements.
Next q lines contains the final arrangement, in the same notation as the initial arrangement.

Output:

Output q lines, "Theft sure!!" or "Theft not sure!!".

Constraints:

Length of initial arrangement will be upto 50. Only slots among the first 50 are used by Digo as per statement. q will be between 1 and 50 both inclusive. Length of final arrangement will be upto 50. Digo finds only the slots among first 50 contains bundles as per statement.

Sample Input:

```
*..*

5

***

.***

*.....*

*

.***.
```

Sample Output:

```
Theft not sure!!

Theft not sure!!

Theft not sure!!

Theft sure!!

Theft not sure!!
```

11. Can you sort them?

Time Limit = 1 sec
Number of Attempts = 248
Number of Correct Solutions = 29

Digo has given you an array of N nonnegative numbers A[1], A[2], ..., A[N]. He considers a nonnegative integer M as good if the sequence A[1] xor M, A[2] xor M, .., A[N] xor M is an increasing sequence. Here, xor is exclusive or, "xor" in Pascal, "^" in C++. Consider the set S of all nonnegative integers M which Digo considers as good.
What is the smallest element in S?

Input
The first line of the input contains an integer T denoting the number of test cases. Then T test cases follow. Each test case has two lines. The first line contains a single integer N. The second line contains N space separated integers A[1], A[2], .., A[N].
Output
Output T lines each containing the required answer i.e., if the set S is empty for the corresponding test case then output "-1"(without quotes) otherwise output the smallest element in S.
Constraints
1 <= T <= 200
1 <= N <= 1000
0 <= A[i] <= 10^18(10 raised to 18)
All A[i]'s will be distinct.

Sample Input
3
4
3 13 8 10
5
10 11 15 5 6
3
2 1 3
Sample Output
4
8
-1
Output Details
In the first test case, If we xor the whole array with 4 then we get 7, 9, 12, 14 which is an increasing sequence. It can be easily seen that 4 is the smallest number. One other possible good number is 20 but no number less than 4 is good.
In the second test case, it can be easily seen that 8 is the required number.
In the third case no nonnegative integer is good hence the answer is -1

12. Can you see them?

Time Limit = 1 sec
Number of Attempts = 30
Number of Correct Solutions = 6

Digo owns a huge fleet of ships. They have been sailing in the Indian ocean from time immemorial. Digo has built 3 lighthouses to guide his ships. Imagine that the layout is the two dimenional cartesian plane and the distance between two points (x1,y1) and (x2,y2) is |x1-x2| + |y1-y2| i.e., we use the Manhattan distance. A lighthouse can see a ship at a distance D if the power in its light is atleast D. Digo wants all his ships to be visible by atleast one of the lighthouses and wants to achieve this by investing as little total power in his lighthouses as possible.

Input:

The first three lines each contain two space separated integers x and y which correspond to the coordinates of the lighthouses. On the fourth line will be a single integer N indicating the number of ships. The following N lines each contain two space separated integers x and y which indicate the coordinates of the ships.

Note that all the lighthouses and ships are on distinct points.

Output:

In a single line, print a single integer P which is the minimum sum total of the powers we need to invest in the lighthouses so that all the ships are visible from atleast one of the lighthouses i.e., if the power invested in the three lighthouses are P1, P2, and P3 in the optimal case then the result you output is P1+P2+P3 i.e., P1+P2+P3 is minimum possible.

Constraints:

1 <= N <= 3000
-10^7 <= x,y <= 10^7

Sample Input:
0 0
3 0
0 3
1
1 1

Sample Output:

2

Explanation:It is easy to see that if we invest power of 2 units in the first lighthouse and 0 in the other two then all the ships are in range.

13. Bribe The President

Time Limit = 1 sec
Number of Attempts = 30
Number of Correct Solutions = 5

Digo works at codevillage, but since he likes playing games, he wants the name to be changed to gamevillage. For this he has to defeat the president of codevillage in a game. The game they planned to play is a 2-player board game, where players get alternate turns. Digo will play the first chance. The game is such that the board contains N blocks numbered from 1 to N, and has exactly one token which will be moved by both players. Initially the token is at position K.

Since everyone at codevillage loves mathematics, they decided to have some mathematics in the game. In a chance, if the token is at block 'A', player may move it to a block 'B' if B >= K and gcd(A,B)=1. Observe that the token can also be moved backwards. To make things interesting, a move from block 'A' to block 'B' would cost (A*B)%M rupees, where M would be decided before starting the game. The first player to make a move to the block N is declared as the winner.

Digo realised that it may not always be possible for him to win the game, as the game may not terminate or his opponent may win. So he bribed the president. Now the president will make the moves as told by Digo. Now of course Digo would have to pay for the moves of both the players. Given the values of N, K and M, your task is to find out whether it is possible for Digo to plan the moves so that he wins. Also if he wins find the minimum money which he will have to spend for the moves of both the players.

Input Specification:

First line consists of three integers N, K and M.
where N = Number of blocks on the board
K = starting block
M = Modulus used in the cost function

Constraints:
2 <= N <= 1500
1 <= K < N
1 <= M <= 3000000

Output:
Output "YES" (without quotes) if it is possible for Digo to plan the moves so that he wins, otherwise output "NO". Also in case of a "YES", print the minimum money which Digo would have to spend in a new line.

Sample Input:
6 2 100

Sample Output:
YES
51

Explanation:
If Digo chooses to move in the following way(2 -> 3 -> 5 -> 6) then using the least amount of money he will be able to finish the game.

Sample Input:
7 2 5

Sample Output:
YES
1

14. Baywatch

Time Limit = 4 sec
Number of Attempts = 6
Number of Correct Solutions = 1

Codevillage is a popular tourist spot. The shape of codevillage is a convex polygon and is surrounded by sea on all the sides and there are many hotels located in it. Digo has become the manager of a Tourism company and believes in providing the best customer service to the tourists who come to his agency. Initially there are N tourists residing in N hotels. There is exactly one tourist in one hotel. Unfortunately, all the tourists got bored of residing in the same hotel so they asked Digo to change their hotels. As Digo is very particular about customer satisfaction he wants to offer them a all-paid complimentary visit to the beach(coastline). For a person to go from (x1,y1) to (x2,y2), the cost of fuel spent is equal to the euclidean distance between them i.e., sqrt((x1-x2)^2 + (y1-y2)^2). He does not want to make his customers happy at the expense of his profits so he wants to change the hotel rooms of the tourists by bearing the minimum cost on fuel. Your task is to give him a rough estimate of the amount he will have to spend on this.

Input:

The first line of each test file contains an integer M which represents the number of segments on the coastline of codevillage. The segments together form a closed convex polygon.

Then the following M lines each contain four space separated integers x1 y1 x2 y2 indicating that a segment of coastline is the line joining the points (x1,y1) and (x2,y2).

Then on a new line a single integer M is given which represents the number of hotels or tourists.

Then the following N lines each contain two space separated integers x y which indicate the coordinates of the hotel. Initially one tourist resides in one hotel.

Output:

Let result be the minimum possible cost of fuel which is required for Digo to change the hotels of each of the tourists. You have to output floor of result i.e., the greatest integer <= result.

Constraints:

3 <= M <= 50 2 <= N <= 50 0 <= x,y,x1,y1,x2,y2 <= 100000

All the hotels are at distinct locations. No segment is repeated in the input. The hotels may be located anywhere on the boundary or in the interior of codevillage.

Sample Input:

```
4
0 0 3 0
3 0 3 3
3 3 0 3
0 3 0 0
```

2
1 1
2 2

Sample Output:

6

Explanation:

Using the basics of geometry we can easily deduce that the minimum cost for each of the 2 tourists to go to the other hotel by visiting the coastline once is sqrt(10). So the total cost is approximately 6.3246 and as we just need a rough estimate so output is the floor of 6.3246 which is 6.