



PROBLEM A
ABNORMAL 90's

A group of n new 90's friends gathered together to celebrate. Now the party is over and they are heading toward home. At each intersection in the path to their houses, they divide into exactly two groups. Size of these two groups can vary depending on the address of their houses. Each people in each group shake hands with all people in the other group and they separate. The two new groups do the same until everybody reaches his/her house. We know that no two people live together.

Now you should compute what will be the maximum number of handshakes in the first separation. (i.e. the first time the group of n people divides) Moreover we want you to count the maximum total number of handshakes in all possible scenarios.

Input (Standard Input)

The input consists of several test cases. For each test case there is an integer n ($1 \leq n \leq 32768$) —the total number of people in the party. The last test case is followed by a line with a single "0".

Output (Standard Output)

For each test output the maximum number of handshakes in the first separation followed by the maximum total number of handshakes in all possible scenarios.

Sample Input and Output

Sample Input	Sample Output
2	1 1
3	2 3
5	6 10
0	



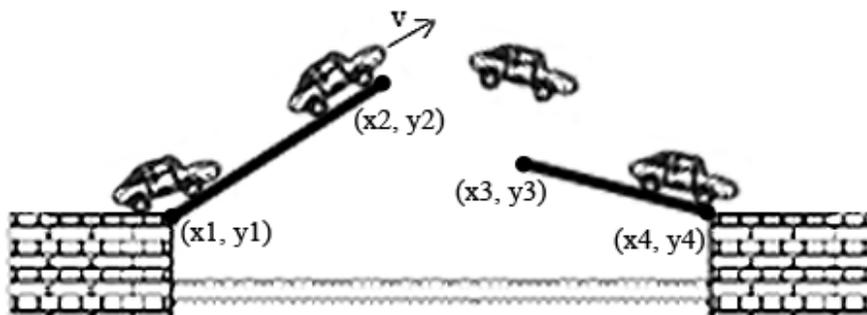
PROBLEM B

BASCULE BRIDGE

“A bascule bridge (sometimes referred to as a drawbridge) is a moveable bridge with a counterweight that continuously balances the span, or leaf, throughout the entire upward swing in providing clearance for boat traffic.”

Wikipedia

A double leaf bascule bridge is constructed on top of a river. But its leaves have different lengths and operate independent of each other. A car on the left side of the bridge wants to cross the bridge. But the bridge is stuck open. Therefore the car has to jump over the river. You can assume that leaves won't move until the car crosses the bridge. The car has a maximum speed of 360 km/h and can be considered as a point. You have to compute minimum possible speed of the car at the upper edge of left leaf such that it can make it over to the other side. The gravity of the earth is 10 m/s^2 .



Double leaf Bascule Bridge

Input (Standard Input)

The input consists of several test cases. The first line of each test case contains integers $x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4$ ($0 \leq x_1 < x_2 < x_3 < x_4 \leq 1000, y_1 = y_4 = 0, 0 \leq y_2, y_3 \leq 1000$) — lower left of left leaf, upper right of left leaf, upper left of right leaf and lower right of right leaf's coordinates respectively. These coordinates are in meters. The last test case is followed by a line with 8 zeros.



Output (Standard Output)

For each test output the minimum speed (in km/h) of the car at the edge left leaf for jumping over to the other side precise to 6 digits after the decimal point. If it is impossible to successfully jump to the other side, print the word "IMPOSSIBLE". (Quotes for clarity)

Sample Input and Output

Sample Input	Sample Output
0 0 3 3 4 0 7 0	5.692100
0 0 3 3 4 3 7 0	11.384200
0 0 3 0 4 3 7 0	IMPOSSIBLE
0 0 1 999 2 1000 3 0	254.558696
0 0 0 0 0 0 0 0	



PROBLEM C

COMPLEX NETWORK

There are n nodes in a computer network, one of which is a server. Nodes are numbered from 1 to n and are connected to each other. To use a link between i -th and j -th nodes we should first initialize it at the cost of $Cost_{i,j}$. After that, each byte we transfer on the link costs 1 unit. The server wants to send a message of length k bytes to all clients of the network. If a node receives the message, it can propagate it to the neighboring nodes. You should compute the minimum cost for this message transfer.

Input (Standard Input)

The input consists of several test cases. The first line of each test case contains integers n, k, s ($1 \leq s \leq n \leq 400, 1 \leq k \leq 10^6$)—the number of nodes in the network (including the server), length of the message and identifier of the server node. Then n lines follow, each consisting of n integers which represent the initialization cost matrix. ($1 \leq Cost_{i,j} \leq 10^6$) It is guaranteed that $Cost_{i,j} = Cost_{j,i}$ and $Cost_{i,i} = 0$ (i.e. a node is immediately connected to itself). The last test case is followed by a line with three zeros.

Output (Standard Output)

For each test output the minimum cost of transferring a message of k bytes from server (node number s) to all the other nodes.

Sample Input and Output

Sample Input	Sample Output
2 3 1	8
0 5	37
5 0	
4 10 3	
0 4 8 1	
4 0 2 5	
8 2 0 20	
1 5 20 0	
0 0 0	



PROBLEM D

DOZENS OF COATS

Cold days of winter are coming and ACM chapter of AUT is getting crowded! There is a coat hanger in the room which is circular and has h columns. Each column of hanger has two *UP* and *DOWN* locations, therefore we can identify a hanging location as a pair (i, UP) or (i, DOWN) where i is the number of a column. This coat hanger has a spinning engine and rotates 1 column a minute in the increasing direction, starting with column 1 in the first minute. At each time unit, people can hang their coats on the UP or DOWN locations of the column that is currently in front or take their coats and leave. We define count function as follows:

- $\text{count}(i, \text{DOWN})$ equals to the number of coats hanging on (i, DOWN)
- $\text{count}(i, \text{UP})$ equals to the number of coats hanging on (i, UP)

People are entering and leaving the room, hanging and taking their coats. i -th person comes at time s_i and leaves at e_i . When someone comes, he hangs his coat on one of the location of the current column, based on the following rules:

- if $\text{count}(\text{current}, \text{UP}) \leq \text{count}(\text{current}, \text{DOWN})$ he hangs his coat on (i, UP)
- otherwise he hangs his coat on (i, DOWN)

When someone wants to leave, he should take his coat first. But there might be a queue of people waiting to take their coats. He does the following:

- While someone in the queue and in front of him has not taken his coat, he waits
- While the current column of the queue differs from the column he hangs his coat on, he waits
- If someone has just arrived, he lets him hang his coat
- While there is a coat on top of his coat, he takes one from top and hangs it on the other location of current column (i.e. from UP to DOWN and vice versa).
- He takes his coat and leaves the queue

You can ignore the time for taking a coat or putting it on another location. Therefore multiple people might take their coats at the same time. You have to simulate this process and for each person output the location he hangs his coat on and the location he takes his coat from. You should also compute the actual time each person leaves.

Input (Standard Input)

The input consists of several test cases. The first line of each test case contains integers n, h ($1 \leq n \leq 1000, 1 \leq h \leq 50$)— the number of people in the group and the number of columns of coat hanger. i -th line of next n lines is a pair s_i and e_i ($1 \leq s_i < e_i \leq 10^4$) — the entering and leaving time for i -th person. No



two persons come or leave at the same time. But one might enter the same time another is going to take his coat and leave. The last test case is followed by a line with two zeros.

Output (Standard Output)

For each test print n lines. For each person (in order they appear in input), output the location he hangs his coat on, the location he takes his coat from and the time he actually leaves the room. See the sample input and output for more clarification.

Sample Input and Output

Sample Input	Sample Output
5 1	(1,DOWN) (1,DOWN) 9
4 9	(1,DOWN) (1,DOWN) 7
2 7	(1,UP) (1,UP) 10
5 10	(1,UP) (1,UP) 8
3 8	(1,UP) (1,UP) 6
1 6	(1,DOWN) (1,UP) 8
4 1	(1,DOWN) (1,DOWN) 7
4 8	(1,UP) (1,DOWN) 6
2 7	(1,UP) (1,UP) 5
3 6	(1,UP) (1,UP) 16
1 5	(1,DOWN) (1,DOWN) 16
4 5	(1,UP) (1,DOWN) 16
1 12	(1,DOWN) (1,UP) 21
6 16	
11 13	
16 17	
0 0	



PROBLEM E

ENCLOSE THE SPY

The wicked spy has stolen some classified information from the central building of investigations and now he is on his way to the exit door. Fortunately federal agents have detected his real identity before he leaves the building and they want to trap him in the building.

The building is in the form of an $n \times m$ grid and is equipped with some invisible walls which can be triggered using a control system. An invisible wall is passable as long as it is not triggered. There is a system to control the invisible walls. At each moment of time, exactly one subset of invisible walls can be triggered. This subset must be chosen from one of the predefined subsets. If the spy is already occupying the location of an invisible wall, triggering that wall has no effect. But still a predefined subset that includes such a wall can be used. (i.e. all the other walls of the subset will be triggered.)

The federal agents failed to use the system properly and the spy is brilliantly passing the walls and getting close to the exit door. So the officials asked you to write a program and determine whether they can trap the spy or not.

In each step the spy can move to one the four adjacent cells (i.e. the cells that share a common edge). Before he makes each move, you can decide which subset of walls to trigger. You may assume that the spy has a full map of the building and knows all predefined subsets. He is clever and always makes the optimal decision.

Input (Standard Input)

The input consists of several test cases. The first line of each test case contains three integers n , m and p ($1 \leq n, m, p \leq 100$) — the number of rows and columns of the building and the number of predefined subset of invisible walls. Each of the next p lines describes a predefined subset. It starts with a positive integer k (≤ 100) — the number of walls in the subset, followed by k pairs of integers — the location of the invisible walls in the subset. All invisible walls in a subset are distinct. The last two lines of each test describe the current location of the spy and the location of the exit door respectively. The spy's current location and exit door are distinct. A location in this problem is represented as a pair r, c ($1 \leq r \leq n, 1 \leq c \leq m$). The input terminates with a line containing three zeros.

Output (Standard Output)

For each test output a line with the word "YES" if you can trap the spy and "NO" otherwise. (Quotes for clarity)

Sample Input and Output

Sample Input	Sample Output
2 2 2	NO



1 2 1	YES
1 1 2	YES
1 1	NO
2 2	
3 4 2	
2 1 3 2 3	
2 3 3 2 3	
2 2	
3 4	
3 3 1	
3 2 1 2 2 2 3	
1 1	
3 3	
3 3 2	
2 2 1 2 3	
1 3 2	
1 1	
3 3	
0 0 0	



PROBLEM F

FOR FUN

For any string s of digits, we generate $f(s)$ as follows:

- break s to its maximal consecutive sequence of equal digits
- for each sequence of digits, write down the digit followed by the number of times it is repeated
- concatenate the string presentation of these numbers in order they appear in s to build $f(s)$

For example $f(111) = 13$ and $f(222122) = 231122$.

Now consider the following definition:

- $f^0(s) = s$
- $f^i(s) = f(f^{i-1}(s))$ for every $i > 0$

Input (Standard Input)

This problem has no input. Use the numerical section of your username of printer as input. You can find it in the problemset package on a piece of paper!

Output (Standard Output)

For each i , $0 \leq i \leq 10$, output $f^i(s)$ on a separate line, which here s is the numerical section of your username of printer. For example if your username is "team12#12345" then $s = 12345$ and your output should be such as sample output.

Sample Input and Output

Sample Input	Sample Output
	12345
	1121314151
	122111311141115111
	112213311341135113
	122211321231411231511231
	.
	.
	.
	.

It is only the first few lines. You should print the whole 11 lines.

Note: if you don't find this problem funny, then it's your own problem. Deal with it!!!



PROBLEM G

GIRAFFE PAINTING

Giraffe wants to repaint the wall of his room. The wall is a sequence of blocks with different shades. Each block has unit length and two adjacent blocks can have the same color. In each step, giraffe puts down his brush somewhere on the wall and paints some consecutive blocks of the wall into some color. Given the current coloring of his wall and coloring he wants to repaint to, compute the minimum number of steps to complete the task.

Input (Standard Input)

The first line of input contains t — the number of test cases. Each test consists of two lines, the current and target colorings of the wall respectively. A coloring of the wall is string of lower English letters. Each letter represents a block of the wall and the letter itself corresponds to the block's shade. For each test, the length of the current coloring equals to the length of target coloring and is at most 200.

Output (Standard Output)

For each test output the minimum number of steps to repaint the wall from the current coloring to the target coloring.

Sample Input and Output

Sample Input	Sample Output
3	1
abcd	2
accc	3
aaa	
pmp	
aaaa	
bcbc	



PROBLEM H

HONOR OF THE MATH

Honor of the math, MJRS, has invented a new operator. For each two positive integers A and B , he defines $A \oplus B = g \times A' \times B'$, such that g is the greatest common divisor of A and B , A' is greatest integer such that A' divides A and it is prime relative to g . B' is defined similarly. More formally:

1. $A' \mid A, B' \mid B$
2. $\text{GCD}(A', g) = 1, \text{GCD}(B', g) = 1$
3. $A' \times B'$ is maximal

For instance $6 \oplus 3 = 3 \times 2 \times 1 = 6$ and $14 \oplus 12 = 2 \times 7 \times 3 = 42$. You can clearly see that the operator is commutative (i.e. $A \oplus B = B \oplus A$).

Given n numbers $a_1 \dots a_n$, choose a subset of it $b_1 \dots b_m$, such that $b_1 \oplus b_2 \oplus \dots \oplus b_m = k$ and m (the count of elements used) is minimum possible.

Input (Standard Input)

The input consists of several test cases. The first line of each test case contains two integers n and k ($1 \leq n \leq 1000, 2 \leq k \leq 10^{12}$) —the number of elements in the array and the target number. The second line of the test contains n integer a_i ($1 \leq a_i \leq 10^{12}$). The last test case is followed by a line with two zeros.

Output (Standard Output)

For each test output the minimum size of a subset of given integers, such that by applying the \oplus operator on the set, we can get k . If no solution can be found, print "NO SOLUTION" on a single line. (Quotes for clarity)

Sample Input and Output

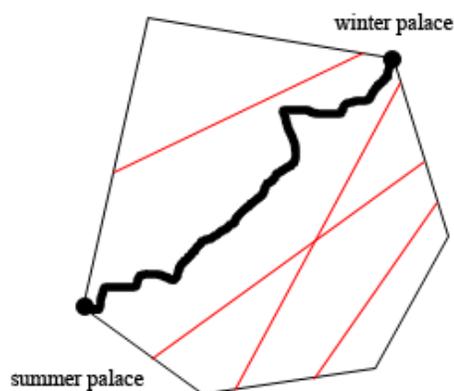
Sample Input	Sample Output
2 6	1
12 6	3
4 900	NO SOLUTION
4 9 25 300	
3 900	
12 45 50	
0 0	



PROBLEM I

IDEAL PATH

The kingdom of heaven is in the form of a convex polygon. The king is migrating from his summer palace to the winter palace. Both palaces are located on the border of the kingdom and specifically on a vertex of polygon. Unfortunately during the last few months, many villagers revolted and claimed their own autonomous territories. The territories are separated with straight lines. You, the wise minister of the king and the leader of royal army have to take precautionary measures for the migration.



Bold line is a secure path inside kingdom of heaven

You know that if the king passes through a different territory from the territory of his summer palace, there is high risk of attacks from angry villagers. There is no way the king leaves the kingdom of heaven. Given the borders of the kingdom and separation lines of territories, you should determine if there is secure path inside the kingdom from the summer palace to the winter palace.

Input (Standard Input)

The input consists of several test cases. The first line of each test case contains two integers n, m ($3 \leq n \leq 1000, 1 \leq m \leq 1000$) — the number of vertices of the convex polygon of the kingdom and the number of border lines. The following n lines describe vertices of the polygon in counter-clockwise order. The description of m border lines follows. For each border line, two distinct points will describe it. Section of the line (that passes through these two points) which lies inside the kingdom is the actual border. This section can have zero length. All coordinates are integers with their absolute values less than or equal to 20000. No border line passes through the summer or the winter palace. The last line of each test case contains two integers s, t ($1 \leq s, t \leq n, s \neq t$)—locations of summer palace and winter palace. Each palace is located exactly on one of the vertices of polygon. The input terminates with a line containing two zeros.



Output (Standard Output)

For each test output a line with the word “YES” if the king can go to the winter palace and “NO” otherwise. (Quotes for clarity)

Sample Input and Output

Sample Input	Sample Output
5 2	YES
0 0	NO
2 0	
4 0	
4 4	
0 4	
2 0 4 3	
5 0 5 1	
1 4	
4 1	
0 0	
2 0	
4 0	
0 4	
2 0 0 4	
1 3	
0 0	



PROBLEM J

JILLION

For any integer x , $\text{rev}(x)$ is the x written in reverse direction. For example $\text{rev}(12234) = 43221$. You are given an integer n , compute the maximum x such that $x - \text{rev}(x) = n$. As the answer can be as large as a jillion and even more, we only consider solutions less than 10^{18} .

Input (Standard Input)

The first line of input contains t ($t \leq 1000$)— the number of test cases. Each test case will be an integer n on a single line ($0 \leq n \leq 10^{18}$).

Output (Standard Output)

For each test output a line containing the maximum x less than 10^{18} such that $x - \text{rev}(x) = n$. If no solution can be found for any integer less than 10^{18} , print "NO SOLUTION" on a single line. (Quotes for clarity)

Sample Input and Output

Sample Input	Sample Output
4	999999999999999999
0	NO SOLUTION
5	98
9	90
81	