Universidad de Buenos Aires

Facultad de Ciencias Exactas y Naturales

Internal selection for the ACM–ICPC 2010 — South American Regional

CONTEST SESSION

September 25, 2010

General Information

Unless otherwise stated, the following conditions hold for all problems.

Input

1. The input must be read from standard input.
2. The input contains several test cases. Each test case is described using a number of lines that depends on the problem.
3. When a line of data contains several values, they are separated by *single* spaces. No other spaces appear in the input. There are no empty lines.
4. Every line, including the last one, has the usual end-of-line mark.
5. The end of input is indicated with a line containing certain values that depend on the problem. This line should not be processed as a test case.

Output

1. The output must be written to standard output.
2. The result of each test case must appear in the output using a number of lines that depends on the problem.
3. When a line of results contains several values, they must be separated by *single* spaces. No other spaces should appear in the output. There should be no empty lines.
4. Every line, including the last one, must have the usual end-of-line mark.
5. No special mark should be written to indicate the end of output.

Problem A
Activities

Ana likes many activities. She likes acrobatics, alchemy, archery, art, Arabic dances, and many more. She joined a club that offers several classes. Each class has a time interval in every week. Ana wants to sign up for many classes, but since they overlap in time, she looks for a subset of non-overlapping classes to attend. A subset is non-overlapping if it does not contain two classes that overlap in time. If a class starts at the time another class ends, this is not considered overlapping.

Ana decided to list all the non-overlapping non-empty subsets of classes. Then she will choose the subset she likes best. In order to predict the amount of paper needed to write the list, she wants you to calculate how many of these subsets there are.

Input

Each test case is described using several lines. The first line contains an integer N indicating the number of classes the club offers ($1 \leq N \leq 10^5$). Each of the next N lines describes a class using two integers S and E that represent the starting and ending times of the class, respectively ($1 \leq S < E \leq 10^9$). The end of input is indicated with a line containing a single -1.

Output

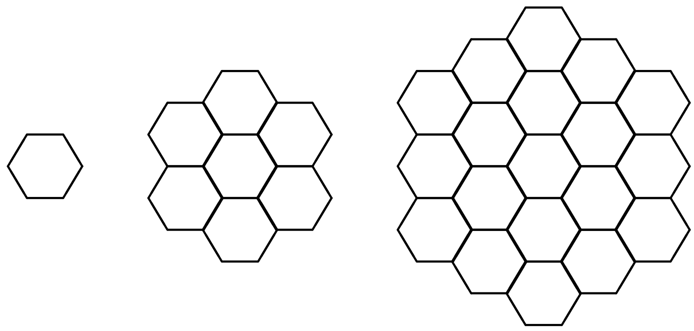
For each test case, output a single line with a single integer representing the number of non-overlapping non-empty subsets of classes. To make your life easier, output only the last 8 digits of the result. If the result has less than 8 digits, write it with leading zeros to complete 8 digits.

Sample input	Output for sample input
5 1 3 3 5 5 7 2 4 4 6 3 500000000 1000000000 1 500000000 1 500000000 1 999999999 1000000000 -1	00000012 00000005 00000001

Problem B
Beehive Numbers

A beehive is an enclosed structure in which some honey bee species live and raise their young. In this problem we consider a two-dimensional sketch of the beehives. Each beehive is composed of a certain number of cells, where each cell is a regular hexagon. Each cell may have some neighbors, which are other cells that share a side with that cell. A cell with exactly 6 neighbors is an internal cell, while a cell with fewer neighbors is an external one. Notice that an external cell can always be changed to internal by adding some neighbor cells.

We are interested in a particular class of beehives. This class of valid beehives is defined recursively as follows: a) a single cell is a valid beehive; and b) given a valid beehive B, if we add the minimum number of cells such that each external cell of B becomes an internal cell, the result is a valid beehive. The figure below shows three valid beehives.



The number of cells in a valid beehive is called a beehive number. Given an integer N, you must decide whether it is a beehive number.

Input

Each test case is described using a single line. The line contains an integer N ($1 \leq N \leq 10^9$). The end of input is indicated with a line containing a single -1.

Output

For each test case, output a single line containing an uppercase "`Y`" if N is a beehive number, or an uppercase "`N`" otherwise.

Sample input	Output for sample input
43 1 7 19 15 -1	N Y Y Y N

Problem C
Camelot

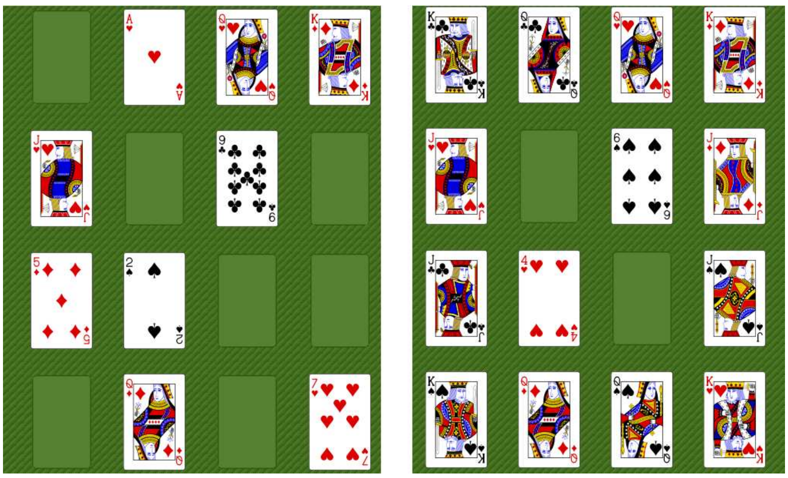
Camelot is a solitaire game that is played with a deck of French cards. The deck contains 52 cards, each of them having a suit and a face value. There are 4 possible suits and 13 possible face values. Since for this solitaire suits are not important, we consider that the deck contains 4 repetitions of each possible face value. Face values are A, 2, 3, 4, 5, 6, 7, 8, 9, 10, J, Q and K.

The solitaire starts with the full deck placed, face down, on the table. There is also a board containing 16 empty slots arranged in a 4 by 4 grid. The game repeatedly alternates two phases: a dealing phase and a removal phase.

The first phase is a dealing phase. During this phase cards are dealt from the deck one at a time. Each card is placed, face up, in an empty slot. However, certain cards can only be placed in specific slots: Jacks (face value J) can only occupy the middle two slots of first and last columns. Queens (face value Q) can only occupy the middle two slots of first and last rows. Finally, Kings (face value K) can only occupy the corner slots. Cards having other face values can be placed in any empty slot. The game is lost whenever a card is dealt from the deck for which no valid empty slot exists. Each time the last empty slot has just been occupied, or when the deck is empty, a removal phase starts.

During a removal phase, it is possible to remove from the board any card or pair of cards that add up to 10. For this purpose, Aces (face value A) are considered as having value 1, while Jacks, Queens and Kings cannot be removed. For instance, it is possible to remove a 10 on its own, a pair formed by a 3 and a 7, a pair formed by an Ace and a 9, etcetera. Cards removed from the board are not used anymore during the game. The removal phase ends when no card can be removed from the board, or when the player decides not to continue removing cards. Notice that it is not mandatory to remove from the board every card that can be removed. However, since the player cannot decide the moment in which a new removal phase will begin, leaving removable cards on the board must be done carefully. Besides, note that if during a removal phase no card is removed, then the game is lost. When the removal phase ends, a new dealing phase starts, unless the deck is empty, in which case the game is over.

The game is won if the deck is empty and only Jacks, Queens and Kings are left on the board.



In the figure above two states of a game are shown. The state on the left represents an ongoing game. Note that Jacks, Queens and Kings are only placed in their corresponding slots, while other cards are placed in any slot. The state on the right depicts an almost finished game. Assuming the deck is empty, the game will be won after the pair formed by the 6 and the 4 is removed.

Camelot is really nice to play, but is frustrating to discover at the end of a game that it was impossible to win because of the initial arrangement of the deck. Even if the initial deck allows the player to win, he may fail to do so because of bad decisions or bad luck when placing or removing cards. Your job in this problem is to find out whether it is at least possible to win the game, given the order in which the cards will be dealt from the deck.

Input

Each test case is described using a single line. The line contains a single string of exactly 52 characters representing the initial arrangement of the deck. The first card dealt from the deck is given by the first character of the string, and so on. Each card is represented by its face value, with the exception of cards with face value 10 that are represented by the digit "`0`". You may assume that the string corresponds to a valid initial arrangement of the deck, i.e., it contains exactly 4 repetitions of each possible face value. The end of input is indicated with a line containing a single asterisk ("`*`").

Output

For each test case, output a single line containing an uppercase "`Y`" if it is possible to win the game with the given initial arrangement of the deck, or an uppercase "`N`" otherwise.

Sample input	**Output**
`AAAA2222333344445555666677778888999900000JJJJQQQQKKKK`	`N`
`JJJJQQQQKKKKA9A9A9A92828282837373737464646465555 0000`	`Y`
`JJJJQQQQKKKKA9A9A9A92828282833337777464646465555 0000`	`N`
`28333377774646464655550000JJJJQQQQKKKKA9A9A9A9282828`	`Y`
`*`	

Problem D
Drawing Quadrilaterals

A quadrilateral consists of 4 points A, B, C and D in the plane, together with the segments AB, BC, CD and DA. Points are called vertices, while segments are called sides. The quadrilateral is simple if opposite sides (i.e., sides that do not share a vertex) do not intersect. Notice that it is possible to have a simple quadrilateral that looks like a triangle, with exactly 3 collinear vertices.

Demetrio has just drawn N points on the wall of his room. He planned to draw a simple quadrilateral having 4 of these points as vertices, and then paint it with blue ink. Demetrio is going to buy the ink right now, but he has not chosen the 4 points yet. Can you tell him the maximum area a simple quadrilateral drawn on his wall can have? In this way Demetrio will be sure he will not run out of blue ink before the work is done.

Input

Each test case is described using several lines. The first line contains an integer N indicating the number of points drawn on the wall ($4 \leq N \leq 1000$). Each of the next N lines describes a different point of the set using two integers X and Y ($-10^7 \leq X, Y \leq 10^7$); these values represent the coordinates of the point in the XY plane. You may assume that within each test case no two points have the same location, neither are all collinear. The end of input is indicated with a line containing a single -1.

Output

For each test case, output a single line with a single decimal number representing the maximum area of a simple quadrilateral having as vertices 4 different points of the input set. Round the result to the closest rational number with one decimal place. In case of ties, round up. Always use exactly one digit after the decimal point, even if it means finishing with a zero.

Sample input	Output for sample input
6 -100 0 100 0 -100 50 0 55 0 -65 1 1 4 -1 0 10000 0 0 0 0 1 -1	12000.0 5000.5

Problem E
Escape from Jail Again

A new International Common Prison for Criminals (ICPC) was built, and your old friend Harry was moved there as a prisoner. As before, the new ICPC is one of the most secure prisons in the world. It was designed by and old gamer and as such, the prison is not necessarily closed, but only an incredibly logical and fast mind can get out.

The new ICPC can be represented as a grid of square cells. Each cell is empty, or it contains a wall, a door, an opening button or a closing button. Harry was accommodated in an empty cell, and all doors were closed. Nevertheless, Harry told you that he will try to escape. Each time Harry is in a cell, he can move in a single step to an adjacent cell (i.e., a cell that shares a side with his current location). Each time Harry steps on a cell that contains an opening button, all doors open, while each time he steps on a cell that contains a closing button, all doors close. Harry can walk around as he wants within the prison, although he cannot move to a cell that contains a wall, neither to a cell that contains a door if the doors are closed.

To escape from the prison, Harry needs to step outside, which means placing himself in one of the cells on the sides and then taking one extra step out in the direction opposite to the prison.

You obtained a map of the prison, and Harry deserves your advise. Tell him the minimum number of steps he needs to escape, or warn him that there is no way to get out.

Input

Each test case is described using several lines. The first line contains two integers N and M indicating respectively the number of rows and columns of the grid that represents the prison ($1 \leq N, M \leq 100$). Line i of the next N lines describes row i of the grid using a string of exactly M characters, where character j represents cell j of that row. This string only contains the following characters with the indicated meanings: "`H`" is the empty cell where Harry is at the beginning; "`.`" is an empty cell where Harry is not at the beginning; "`W`" is a wall; "`D`" is a door; "`O`" is an opening button; and "`C`" is closing button. You may assume that within each test case there is exactly one character "`H`". The end of input is indicated with a line containing the number -1 twice.

Output

For each test case, output a single line with a single integer representing the minimum number of steps Harry needs to escape the prison, or the number -1 if it is impossible for him to do so.

Sample input	**Output for sample input**

`5 8`
`WWWWWWW.`
`WHDC...D`
`W.WW.WCW`
`W.OW..OW`
`.WWWWWWW`
`3 3`
`ODO`
`DHD`
`ODO`
`3 7`
`WWWWWWW`
`DH..OCD`
`WWWWWWW`
`4 1`
`W`
`H`
`O`
`W`
`1 13`
`HOW.DO.COW.DO`
`-1 -1`
	`21`
	`-1`
	`8`
	`1`
	`1`

Problem F
File Recovery Testing

In a recent programming contest appeared a problem named "File Recovery". In that problem, repeated strings of a given text were to be counted. You are preparing test cases for that problem, and in order to test for border cases you want to generate a text with many repetitions of a particular string.

Of course, test cases cannot be arbitrarily long, so you decided to choose a length and a string, and then fit in a text of that length as many repetitions as possible of the string. For instance, if the length is 14 and the string is "`abcab`", you may generate the text "`abcabcabcabcab`" whose length is 14 and where the string "`abcab`" appears 4 times (starting at positions 1, 4, 7 and 10).

You would like to know how good your idea is before implementing. Given a length and a string, you must determine the maximum number of times the characters of the string can appear consecutively in a text of that length.

Input

Each test case is described using a single line. The line contains an integer K ($1 \leq K \leq 10^9$) and a non-empty string S of at most 10^6 lowercase letters. The end of input is indicated with a line containing the number -1 and an asterisk ("`*`").

Output

For each test case, output a single line with a single integer representing the maximum number of times the characters of S can appear consecutively in a text of length K.

Sample input	**Output for sample input**
`14 abcab`	`4`
`1000 abcde`	`200`
`1000000000 z`	`1000000000`
`1 zzzzz`	`0`
`-1 *`	

Problem G
Girls and Boys

There are G girl students and B boy students in a class that is about to graduate. You need to arrange them in a single row for the graduation. To give a better impression of diversity, you want to avoid having too many girls or too many boys seating consecutively.

You decided to arrange the students in order to minimize the gender regularity. The gender regularity of an arrangement is the maximum number of students of the same gender (all girls or all boys) that appear consecutively.

Given G and B, calculate the minimum gender regularity among all possible arrangements.

Input

Each test case is described using a single line. The line contains two integers G and B representing the number of girls and boys in the class, respectively ($0 \leq G, B \leq 1000$). The end of input is indicated with a line containing the number -1 twice.

Output

For each test case, output a single line with a single integer representing the minimum gender regularity that an arrangement of G girls and B boys can have.

Sample input	**Output for sample input**

`10 10`
`5 1`
`0 1000`
`-1 -1`
	`1`
	`3`
	`1000`

Problem H
Hackers

The network of your office is composed of several computers and bidirectional links. Each link connects a given pair of computers and has an access value. Each user in the network has an access privilege, and is able to use any link with access value not exceeding his access privilege.

Everything was fine until suddenly you notice that a bunch of hackers are accessing the network. You know that if there is a link between computers A and B, with access value V, and a hacker with access privilege of at least V controls A, then he can control B. Hackers wish to control the most important computers by exploiting problems in the network. They are trying to increase their access privileges in order to use the links, and your task is to measure how safe the network is.

Given the description of the network, the computer each hacker currently controls and the target computer each hacker wishes to control, you need to calculate the minimum access privilege each hacker needs to get in order to control his target computer.

Hackers act independently, neither they collaborate nor interfere with each other. This means that each hacker may control each computer and use each link independently of what the other hackers do.

Input

Each test case is described using several lines. The first line contains three integers C, L and H, indicating the number of computers, links and hackers in the network, respectively ($2 \leq C \leq 3000$, $1 \leq L, H \leq 10^5$); each computer is identified by an integer number between 1 and C. Each of the next L lines describes a different bidirectional link using three integers A, B and V; the numbers A and B identify two distinct computers that are the endpoints of the link ($1 \leq A < B \leq C$); the number V is the access value of the link, that is, any hacker must have an access privilege of at least V to use the link ($1 \leq V \leq 10^9$). Each of the last H lines describes a different hacker using two distinct integers S and T that identify the computer that the hacker currently controls and the computer that the hacker wishes to control, respectively ($1 \leq S, T \leq C$). You may assume that within each test case no two links connect the same pair of computers, and that for any pair of computers there is at least one sequence of links that allow to reach one computer starting from the other. The end of input is indicated with a line containing the number -1 three times.

Output

For each test case, output a single line with H integers representing the minimum access privilege each hacker needs to achieve its goal. The result for each hacker must appear in the same order that the hackers are described in the input.

Sample input	Output for sample input
5 6 4 1 2 4 1 3 5 2 4 3 2 5 1 3 4 2 4 5 2 3 2 2 4 1 5 3 1 2 1 1 1 2 1 2 1 2 1 3 1 2 1000000000 2 1 2 1 1 2 -1 -1 -1	2 2 4 4 1 1000000000 1000000000 1000000000

Problem I
Imperial Units

As you may know, there are currently two main sets of measurement units in the world: the metric system and the imperial system. The imperial system receives its name from the British empire, which was the place of its invention and its main user until recently. Nowadays, Britain's heir, the United States of America, is the only country where a variation of the imperial system is the official measurement system.

For a particular magnitude, in a given measurement system there are N different units $U_1, U_2, \ldots U_N$ (the number of units depends on both the magnitude and the system). For every i $(1 \leq i \leq N-1)$, a certain number of U_i is equivalent to a certain number of U_{i+1}. In the metric system we always have that $\mathbf{1}U_i$ is equivalent to $\mathbf{10}U_{i+1}$. For instance, 1 decimeter is equivalent to 10 centimeters, 1 gram is equivalent to 10 decigrams, and 1 decaliter is equivalent to 10 liters. On the contrary, in some variations of the imperial system we may have other positive integers instead of 1 and 10. For instance, 32 drams are equivalent to 875 grains.

Since you were born and raised using the much more sensible metric system, you need help learning the imperial system and its variations. You want to be able to transform directly from U_1 to U_N, that is, you need to know that a certain number of U_1 is equivalent to a certain number of U_N. To ease further calculations, you want to express the equivalence using only integers values, and these values must be as small as possible.

Input

Each test case is described using several lines. The first line contains an integer N indicating the number of units in the measurement system $(2 \leq N \leq 10)$. Line i of the next $N-1$ lines describes the relationship between units U_i and U_{i+1} with two integers A_i and B_i representing that A_iU_i is equivalent to B_iU_{i+1} $(1 \leq A_i < B_i \leq 100)$. The end of input is indicated with a line containing a single -1.

Output

For each test case, output a single line with two positive integers C and D representing that CU_1 is equivalent to DU_N. If there are several alternatives, choose the minimum possible value for C.

Sample input	**Output for sample input**
5 1 2 2 3 3 4 2 5 2 6 9 -1	1 10 2 3

Problem J
Jara's Legacy

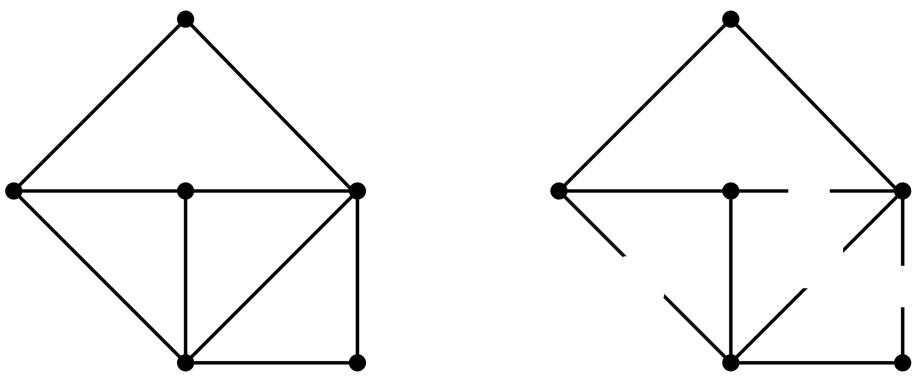
Victor Jara was a Chilean teacher, theater director and political activist. He is widely recognized because of his talent as poet and song writer. His most recognized work is probably the song "A Desalambrar" that can be translated from Spanish as "Unwire". In this song Jara assures that people is the rightful owner of the lands, and so wire fences that delimit private properties should be cut down to allow access to everybody.

Although Jara's proposal is far from being fulfilled, some of his convinced listeners keep trying to make it happen. Since they must face several enemies, they try to make their job efficient by only cutting down the necessary number of fences and not more.

Each fence can be modeled as a segment (straight line) connecting two points in the XY plane. These endpoints are considered to be part of the fence. A cut in a fence removes any contiguous part of the fence except the endpoints.

An area is said to be free if and only if, for any pair of points not lying over a fence, there is a (not necessarily straight) line that connect these points without crossing any fence.

In the following figure you can see a possible scenario with 9 fences (endpoints were represented using small circles). The picture on the left depicts the original fence distribution; the area is not free because there are pairs of points that cannot be connected without crossing the fences. The picture on the right shows that 4 cuts are enough to obtain a free area.



Given the location of the fences, your job is to calculate the minimum number of fences that need to be cut down to make the area free, according to the above definition.

Input

Each test case is described using several lines. The first line contains an integer N indicating the number of fences in the area ($1 \leq N \leq 10^5$). Each of the next N lines describes a different fence using four integers X_0, Y_0, X_1 and Y_1 ($-10^4 \leq X_0, Y_0, X_1, Y_1 \leq 10^4$). These values represent that there is a fence whose endpoints in the XY plane are (X_0, Y_0) and (X_1, Y_1). You may assume that for each fence its two endpoints are distinct. Besides, within each test case, the intersection of any pair of fences is either empty or it is an endpoint of both fences. The end of input is indicated with a line containing a single -1.

Output

For each test case, output a single line containing a single integer representing the minimum number of fences that need to be cut down to make the area free.

Sample input	Output for sample input
9 -50 0 0 0 0 0 50 0 -50 0 0 50 0 50 50 0 -50 0 0 -50 0 -50 50 0 0 0 0 -50 0 -50 50 -50 50 -50 50 0 2 0 1 2 3 0 0 2 2 -1	4 0