Problem A: Pie Delivery

Filename: `delivery`
Timelimit: `10 seconds`

Apple Bloom hates being babied and wants to impress her sister. Apple Bloom will deliver pies to a neighboring town, by herself, once and for all showing how responsible she is. Luckily Apple Bloom has borrowed a map from a local librarian that details the distance and locations of various geographical features that can be used as a reference. Apple Bloom has now constructed a list of times she expects it to take to travel between various locations of the map. She has ordered each of the locations numerically and to ensure she's making progress towards her final destination, she always moves from any location to another one with a higher numerical value.

Apple Bloom wants to finish the delivery as fast as possible, so please write a program that will take in a description of the travel times and return the shortest time that the delivery can be completed in. Apple bloom will always start at her farm (location 1) and end at the delivery point (location ***n***).

Input

The first line of input will be a single integer ***n*** ($2 \le n \le 500$), which represents the number of locations on the map. The following ***n*** lines contain ***n*** space separated integers, representing the distance to traverse a particular portion of the trip. More specifically, the j^{th} value on the i^{th} of these lines will represent the time units taken to move from location ***i*** to location ***j***. If an integer is -1 then that path was considered infeasible. Each valid move will take in between 1 and 1000 time units, inclusive. (Note: it is guaranteed that all travel paths from to ***i*** to ***j*** with $i \geq j$ will be infeasible. It is also possible that some other travel paths are infeasible.)

Output

Output on a line by itself, the shortest number of time units to move from location 1 to location ***n***. There will always be a feasible way to get from the farm to the delivery point.

Samples

Input	Output
4 -1 5 2 8 -1 -1 1 10 -1 -1 -1 5 -1 -1 -1 -1	7

Problem B: Connect the Dots

Filename: `dots`

Timelimit: `2 seconds`

To practice her numbers, Arup's daughter Anya has recently started playing connect the dots. Normally, one connects a dot with the number n to the dot with the number n+1, creating one connected drawing. Unfortunately, Anya is very stubborn and she chooses whichever pair of dots she wants to connect. Sometimes, Anya's creation becomes several separate connected drawings. Given a list of which dots Anya has connected, write a program that counts the number of separate connected drawings she has. Each connected drawing must have at least one line between some pair of dots in it.

Input

The first line of input contains two positive integers, ***d*** (***d*** ≤ 1,000), representing the number of dots in the drawing, and ***c*** (***c*** ≤ 10,000), representing the total number of lines between pairs of dots Anya has decided to draw. The following ***c*** lines contain two distinct positive integers representing a single line between two dots. The stars in each test case will be numbered 1 through ***s***, inclusive. A connection is considered bidirectional, thus, if a is connected to b, b is connected to a.

Output

On a line by itself, output the number of different connected drawings Anya has formed.

Samples

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

Problem C: Lure Module

Filename: `lure`

Timelimit: `2 seconds`

Pokemon Go has taken over Treetopia! The mayor has decided to hold a weekend event placing lure modules in pokeestops throughout Treetopia so that all trees are covered by a lure module!

All trees in Treetopia have a pokeestop and the lure module will cover the area of the tree itself as well as any other tree directly connected by a bridge. Determine the minimum number of lure modules needed to cover all trees in Treetopia!

Input

The first line of input is an integer n ($1 \leq n \leq 10^5$), representing the number of trees in Treetopia. The next n-1 lines contain two integers a and b ($1 \leq a, b \leq n$), meaning that trees a and b are connected by a bridge. Treetopia is connected. In other words, anyone in a tree in Zootopia can go anywhere else by traveling several bridges.

Output

Output a single integer, the minimum number of lure modules required to cover treetopia.

Samples

Input	Output
4 1 2 1 3 1 4 1 5	1
6 1 2 2 3 3 4 4 5 5 6	2

Problem D: Prank War

Filename: `prankwar`
Timelimit: `3 seconds`

Jacob and Maxwell have been involved in quite the prank war over the past couple of days. Maxwell wants to continue to pranking Jacob using empty water bottles. He has a ton of water bottles and has planned out a large collection of pranks. Each prank, however, uses a different number of water bottles. He wants to know what the maximum amount of pranks he can do using the water bottles he has.

Maxwell can only do each prank up to one time.

Input

The input will consist of two positive integers n ($1 \le n \le 1,000$) and w ($1 \le w \le 100,000,000$), representing the number of prank ideas and water bottles Maxwell has, respectively. The next n lines each contain a single integer c ($0 \le c \le 1,000,000$) representing the number of water bottles that prank uses.

Output

Output a single line consisting of a single integer representing the maximum amount of pranks Maxwell can pull on Jacob using his water bottles.

Samples

Input	Output
5 10 4 8 2 4 2	3

Problem E: Simi at Starbucks

Filename: `starbucks`
Timelimit: `3 seconds`

When Arup's daughter, Simi, wasn't lecturing the class about recycling or not giving up, she was begging Arup to take her to Starbucks. Whenever Simi goes to Starbucks, she buys some combination of two items: Frappuccinos and cake pops. Recently, Simi got the Starbucks app and has some money loaded onto it. She insists on not wasting any of the money and wants to spend all of it, to the penny. Given the cost of a Frappuccino, the cost of a cake pop, and how much money she has to spend, help her determine how many different combinations of the two items she can buy to spend all of her money.

Input

The input will consist of three space-separated positive integers: f ($100 \le f \le 1000$), the cost of a single Frappuccino in cents, c ($100 \le c \le 1000$), the cost of a cake pop, in cents, and m ($100 \le m \le 10^9$), the number of cents she has to spend at Starbucks.

Output

On a line by itself, output the number of different ordered pairs (a, b) such that a Frappucinos and b cake pops cost precisely m cents.

Samples

Input	Output
145 200 345	1
200 100 600	4
100 100 1000000000	10000001

Problem H: Headache

Filename: `headache`
Timelimit: `2 seconds`

Lillian's class of ***N*** children is about to go on a field trip to Seaworld to see Shamu for the last time before he "retires", as told to the kids. Since the class is so big it can be difficult to chaperone the whole group together so Lillian wishes to split the class into two groups. The problem is some children frequently argue and gets into fights with each other and if they were in the same group this could cause problems. This causes big headaches for Lillian. She wishes to minimize her headaches and needs your help to figure out the best split of children such that the largest conflict between any pair of children in the same group is minimized. (For example, if one split causes conflicts of size 5, 8 and 12 in group 1 and conflicts of size 4, and 7 in group 2, Lillian's headache is of size 12, the maximum of all the conflicts. Alternatively, if a different split causes conflicts of size 6, 7 and 9 in group 1 and conflicts of size 4 and 8 in group 2, then Lillilan's headache is of size 9 and this second arrangement is preferable to the first.)

Input

The first line contains two integers N $(1 \le N \le 20000)$ and M $(0 \le M \le 20000)$ which represent the number of children and number of pairs of children that fight respectively. The next M lines each describe the i^{th} pair of children who fight and how much of a headache this will cause Lillian. Each line contains three integers U_i, V_i, $(1 \le U_i, V_i, i \le N, U_i \ne V_i)$ and $W_i (1 \le W \le 10000)$. U_i and V_i will be a pair of children who will get into a fight and W_i is value of the headache that Lillian will get if they fight. You are guaranteed that each pair of children who fight in the input will be distinct. (Thus, if one pair of children is 5 and 7, a future pair of children can not be 5 and 7 or 7 and 5.)

Output

Output the minimum possible size of Lillian's headache over all possible ways to split the children into two groups. If there is a way to partition the class that causes no headaches, output 0.

Samples

Input	Output
2 1 1 2 1	0
3 3 1 2 5 2 3 4 1 3 3	3