,

Problem A. Yuriy and training 1

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	256 megabytes

Yuriy decided to become more fit. To achieve it, he will run from home to work every morning.

The road system in the Kyiv consists of n nodes and m roads. Every road connects two nodes. Two roads can't connect the same pairs of nodes. There is no road that connects a node with itself.

There is a supermarket on every road, where you can buy RedCat. This is an energy drink, without which Yuriy can't cross the road. When Yuriy is crossing the road he buys exactly one can of RedCat. There is a limited number of RedCat in every supermarket. When there is no way from home to the work, each road of which contains at least one RedCat, Yuriy will stop his training and start working from home.

Input

First line contains two numbers $2 \le n \le 500$, $0 \le m \le 1000$. Next m lines contain 3 numbers $1 \le u \le n$, $1 \le v \le n$, $1 \le a \le 10^6$ - nodes connected by this road and the number of RedCat on it. Yuriy lives in node number 1 and MemSQL office is located in the node n.

Output

Output number of days when Yuriy can run from home to work.

,

Example

standard input	standard output
10 30 5 6 204308 5 4 551943 6 7 418130 9 10 787536 7 5 502144 6 4 888746 6 2 503442 2 7 656246 8 1 381104 9 8 609745 3 1 250504 8 4 387317 4 1 110701 3 4 912029 1 7 958718 8 3 2340 3 10 50904 2 9 239079 2 5 496036 5 3 303751 2 1 73817 2 8 100847 8 7 474795 4 9 602761 6 1 181411 7 4 513869 7 10 921279 5 10 872200 3 6 295463 9 1 445908	2402163

,

Problem B. Yuriy and training 2

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

The road system in Kyiv was changed. Now it contains more roads and nodes. Yuriy needs your help !!!!

Yuriy decided to become more fit. To achieve it, he will run from home to work every morning.

The road system in the Kyiv consists of n nodes and m roads. Every road connects two nodes. Two roads can't connect the same pairs of nodes. There is no road that connects a node with itself.

There is a supermarket on every road, where you can buy RedCat. This is an energy drink, without which Yuriy can't cross the road. When Yuriy is crossing the road he buys exactly one can of RedCat. There is a limited number of RedCat in every supermarket. When there is no way from home to the work, each road of which contains at least one RedCat, Yuriy will stop his training and start working from home.

Input

First line contains two numbers $2 \le n \le 1000$, $0 \le m \le 3000$. Next m lines contain 3 numbers $1 \le u \le n$, $1 \le v \le n$, $1 \le a \le 10^5$ - nodes connected by this road and the number of RedCat on it. Yuriy lives in node number 1 and MemSQL office is located in the node n.

Output

Output number of days when Yuriy can run from home to work.

Example

standard input	standard output
10 30 5 6 6036 5 4 22446 6 7 79355 9 10 89573 7 5 92430 6 4 63382 6 2 48952 2 7 3836 8 1 85365 9 8 74586 3 1 87331 8 4 36512 4 1 90720 3 4 3754 1 7 85815 8 3 41126 3 10 77880 2 9 15790 2 5 6036 5 3 1037 2 1 55617 2 8 22510 8 7 85203 4 9 11014 6 1 4363 7 4 30745 7 10 28711 5 10 75658 3 6 78789 9 1 75472	271822

,

Problem C. Yuriy and training 3

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	256 megabytes

The road system in Kyiv was changed. Now it contains more roads and nodes. Now the city is a planar graph. Yuriy house is one of the nodes with the smallest x coordinates when the MemSQL office is one of the nodes with the greatest x coordinates. Yuriy needs your help !!!

Yuriy decided to become more fit. To achieve it, he will run from home to work every morning.

The road system in the Kyiv consists of n nodes and m roads. Every road connects two nodes. Two roads can't connect the same pairs of nodes. There is no road that connects a node with itself.

There is a supermarket on every road, where you can buy RedCat. This is an energy drink, without which Yuriy can't cross the road. When Yuriy is crossing the road he buys exactly one can of RedCat. There is a limited number of RedCat in every supermarket. When there is no way from home to the work, each road of which contains at least one RedCat, Yuriy will stop his training and start working from home.

Input

The first line contains one numbers $2 \leq n \leq 10^5$. Next n lines contain the coordinates of the nodes $0 \leq x_i \leq 10^5, 0 \leq y_i \leq 10^5$. Next m lines contain 3 numbers $1 \leq u \leq n$, $1 \leq v \leq n$, $1 \leq a \leq 10^5$ - nodes connected by this road and the number of RedCat on it. Yuriy lives in node number 1 and MemSQL office is located in the node n. It is guaranteed that roads are not intersecting.

Output

Output number of days when Yuriy can run from home to work.

,

Example

standard input	standard output
10 23333 92971 32760 90408 36756 68794 92121 61328 59613 23256 56694 97455 62500 41662 57882 92745 39825 60628 95092 85360 21 5 7 907 9 8 277 7 9 204 7 8 337 7 4 966 3 2 840 3 8 740 4 8 700 2 8 684 8 6 33 10 8 77 9 3 709 3 1 845 2 1 576 2 6 862 5 9 325 9 1 914 1 6 283 10 6 431 4 10 837 5 4 193	1345

,

Problem D. lunch

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	256 megabytes

People from MemSQL eat their food in the office. There are m people working in MemSQL. For every one of them, it is known how many grams of food he can eat in an hour s_i.

About every dish, it is known that it will be ready at time r_i and it will be fresh until d_i, and its weight is p_i.

Workers will eat all the lunch. At the same time, two workers can't eat the same dish, but they can swap dishes at any time.

Suppose e_i - the time when i-th dish was eaten. Workers will eat the food in the way, that the value $\max_{1 \le i \le m} \max(0, e_i - d_i)$ is minimum possible.

Input

The first line contains two numbers $n, m (1 \le n \le 30, 1 \le m \le 30)$. Next n lines contain 3 numbers $p_i, r_i, d_i (1 \le p_i \le 10^5, 0 \le r_i l d_i \le 10^7)$. Next m lines contain one number$s_j (1 \le s_j \le 10^5)$.

Output

Output the minimum value of $\max_{1 \le i \le m} \max(0, e_i - p_i)$ with precision better then 10^{-4}

Examples

standard input	standard output
2 2 13 0 4 10 1 3 4 2	0.4999974
1 1 1 0 1 1	0.0000000
2 2 10 0 1 10 0 1 6 4	0.9999976

,

Problem E. chess

Input file:	standard input
Output file:	standard output
Time limit:	1 second
Memory limit:	256 megabytes

You have a chessboard of the size $n \times m$. Some squares on this board are forbidden. You need to place maximum number of horses on this board. There should not be a pair of horses that can eat each other.

Input

The first line contains two numbers $1 \le n \le 100$, $1 \le m \le 100$. The second line contains one number - count of the forbidden squares $0 \le k \le n \times m$. Next k lines contain coordinates of the forbidden squares by two numbers per line, $1 \le x \le n$, $1 \le y \le m$

Output

Output the maximum number of horses you can place in this board.

Examples

standard input	standard output
8 8 0	32
8 8 2 1 1 2 2	32

,

Problem F. stas birthday

Input file:	standard input
Output file:	standard output
Time limit:	1 second
Memory limit:	256 megabytes

Stas got an undirected graph as a present for his birthday. He hates even numbers. He wants to delete some edges, so the degree of every node becomes odd. Please, help Stas.

Input

The first line contains two numbers $1 \leq n \leq 2 * 10^5$, $0 \leq m \leq 2 * 10^5$ - number of nodes and number of edges. Next m lines contain information about the edges - $1 \leq u \leq n$, $1 \leq v \leq n$ (nodes connected by this edge).

Output

If it is impossible - output -1. Otherwise, output the number of edges that will not be deleted. On the second line output indexes of these edges. (indexes are started from 1).

Examples

standard input	standard output
2 1 1 2	1 1
4 4 1 2 2 3 3 4 4 1	2 1 3
3 1 1 2	-1

,

Problem G. locating

Input file:	standard input
Output file:	standard output
Time limit:	1 second
Memory limit:	256 megabytes

Kyiv office of the MemSQL moves to the new room. Vova wants to locate workers in this room.

The room consists of the raw with tables. Every worker has his best friend. If the best friend of worker x is worker y, then the best friend of the worker y is a worker x. So all workers are split into the pairs.

Vova wants the minimum distance between best friends to be as big as possible (otherwise they will speak with each other instead of working). From all possible ways to locate workers where this distance is maximum, he will choose the one, where the value $\sum_{i=1}^{2n} |P_{old_i} - P_{new_i}|$ will be minimum (P_{old_i} — the old place of the worker i, P_{new_i} — the new place of the worker i).

Input

First line contains the count of best friends pairs n ($1 \le n \le 100$). The next line contains $2n$ numbers, that explains the relationship between workers. If $a_i = a_j$ ($1 \le a_k \le n$), then i-th and j-th workers — are best frineds.

Output

Output $2 * n$ numbers - new locating of the workers in the same format as in the input.

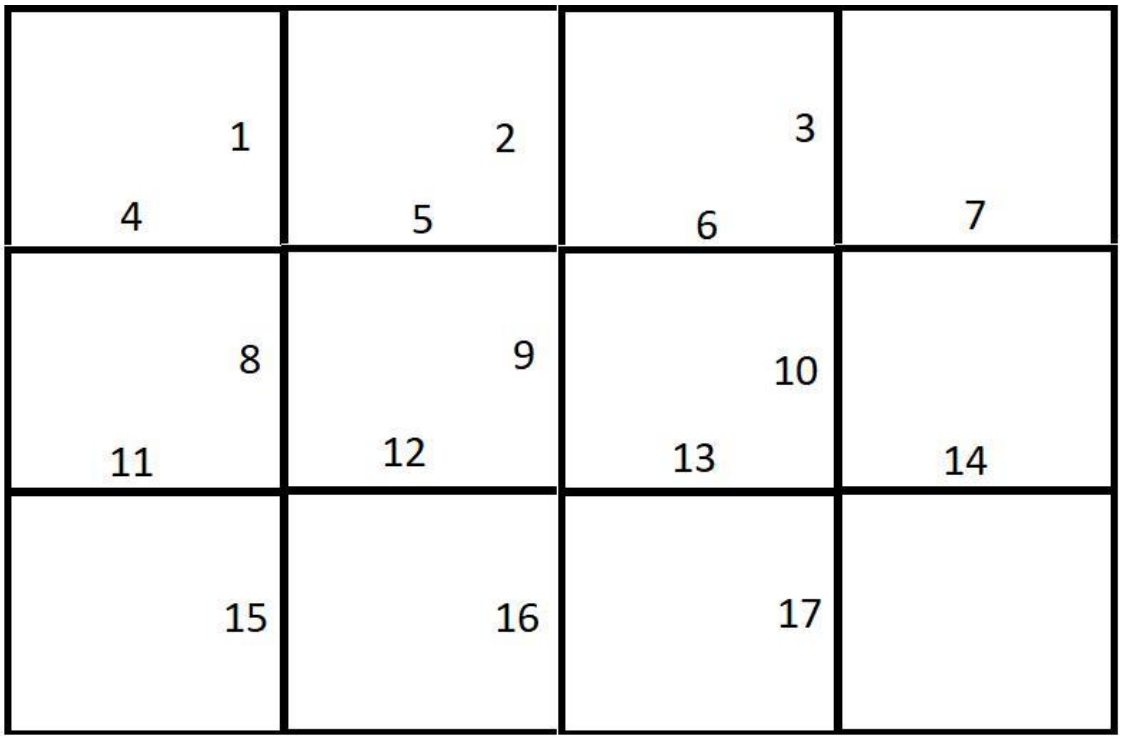
Examples

standard input	standard output
4 1 3 2 2 1 4 4 3	1 2 4 3 1 2 4 3
1 1 1	1 1
2 1 1 2 2	1 2 1 2

Problem H. game

Input file:	standard input
Output file:	standard output
Time limit:	1 second
Memory limit:	256 megabytes

Archie decided to play a game. He has a $3 * 4$ field. He can move between squares that have the common side. There are coins on the square sides. When Archie moves through side he takes $\lceil x/2 \rceil$ coins. If there are no more coins on the side, then Archie can't move through it. Archie can start at any square and end his travel at any square. Help Archie to get the maximum number of coins.



Input

The input contains of 5 lines. First, third and fifth lines contain 3 numbers - the count of coins on the vertical sides. The second and fourth lines contain 4 numbers - the count of coins on the horizontal sides. All numbers are positive and not bigger than 10^9.

Output

Output one number - the maximum number of coins Archie can get.

Examples

standard input	standard output
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17	150
1 1 1 0 0 0 0 1 1 1 0 0 0 1 1 1 1	7