Belgium Algorithm Contest

Belgium Algorithm Contest
Round 1 - 2018

Do not open before the start of the contest.

This page is not really blank.

Belgium Algorithm Contest

Problem A
Grid Paths
Time limit: 1s

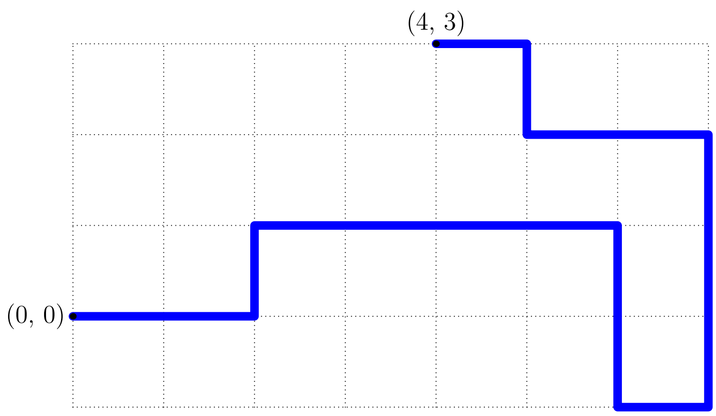
Gridland is a very strange land where all the roads form a giant grid. In Gridland there is a delicious pizzeria conveniently located at the $(0,0)$. A customer can order a pizza to be delivered at any location (x,y) with integer coordinates in Gridland. Customers in Gridland are very picky. When they order a pizza they want it to be delivered exactly T minutes later, even if that means that the pizza will be cold. The pizzeria chef cannot stand to watch his delicious pizzas getting cold so he sends them out for delivery as soon they are ordered.

In Gridland, vehicles have to travel at constant speed and are not allowed to visit the same intersection twice in the same travel. It takes exactly one minute to go from one intersection to the next.

Given the location of a customer (x,y), and T can you help the pizzeria figure out whether it is possible to deliver a pizza exactly T minutes after it was ordered? Assume that the pizzas are instantly ready.

Example

If the customer is at $(4,3)$ and $T = 17$ then it is possible to deliver the pizza as follows:



Input

The first line contains two integers x and y, the coordinates of the customer.

The second line contains the integer T, the exact time at which the pizza needs to be delivered to the customer.

Constraints

- $-10^6 \le x, y \le 10^6$
- $1 \le T \le 10^6$

Output

A single line with `possible` if it is possible to deliver the pizza exactly T minutes after it was ordered or `impossible` otherwise.

Sample test cases

Input 1

```
4 3
17
```

Output 1

```
possible
```

Input 2

```
4 3
14
```

Output 2

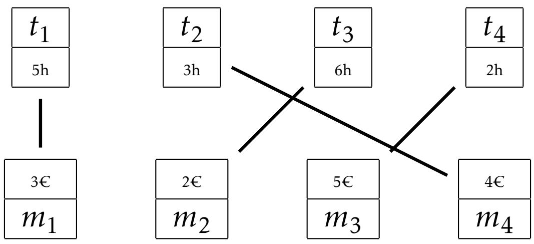
```
impossible
```

Problem B
Task assignment
Time limit: 3s

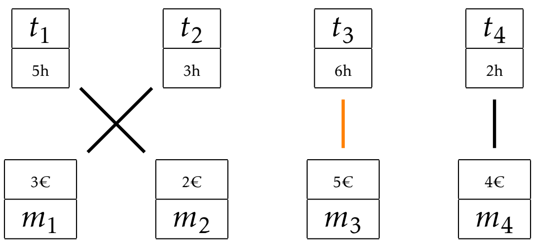
You have n tasks to perform. Each of them has a duration in hours d_i. In order to do them, you will rent n machines and assign one task to each of the machines. Each machine will run exactly one task and tasks cannot change machines. Each machine costs c_i euros per hour. You will receive q queries each with a task index t and machine index m. For each query we are interested to know the cost incurred by forcing task t to be performed on machine m while assigning the other in the best way possible.

Example:

Suppose that $n = 4$ with $d = [5, 3, 6, 2]$ and $c = [3, 2, 5, 4]$. The minimum cost to perform all the tasks is $5 \times 3 + 3 \times 4 + 6 \times 2 + 2 \times 5 = 49$ with the assignment shown in the figure.



Suppose that $q = 1$ and the query is $t = 3, m = 3$. This means that task 3 must be performed on machine 3. The minimum cost to achieve this is $5 \times 2 + 3 \times 3 + 6 \times 5 + 2 \times 4 = 57$ as shown in the figure.



Thus the cost incurred of forcing task 3 on machine 3 is $57 - 49 = 8$.

Input

The first line contains a single integer n giving the number of tasks.

The second line contains n integers giving the task durations d_i.

The third line contains n integers giving the hourly prices of the machines c_i.

The next line contains a single integer q giving the number of queries.

Then follow q lines. The i-th contains two integers t_i and m_i giving the index of the task and machine as described above.

Constraints

- $2 \leq n \leq 10^5$
- $1 \leq d_i \leq 10^5$
- $1 \leq c_i \leq 10^5$
- $1 \leq q \leq 10^5$
- $0 \leq t_i, m_i < n$

Output

Output q lines. The i-th line must contains a single integer giving the cost incurred by forcing task t_i on machine m_i.

Note that queries are independent. So each cost is computed relative to the minimum cost achievable when no restrictions are made.

The cost can be very big, use long for your calculations.

Sample test cases

Input 1

```
5
4 3 2 4 5
1 4 3 2 1
4
0 4
1 3
2 3
4 4
```

Output 1

```
0
1
3
0
```

Input 2

```
6
100 143 67 27 32 44
75 302 102 33 12 88
3
0 4
1 5
3 4
```

Output 2

```
903
5382
7260
```

This page is not really blank.

Belgium Algorithm Contest

● Problem C

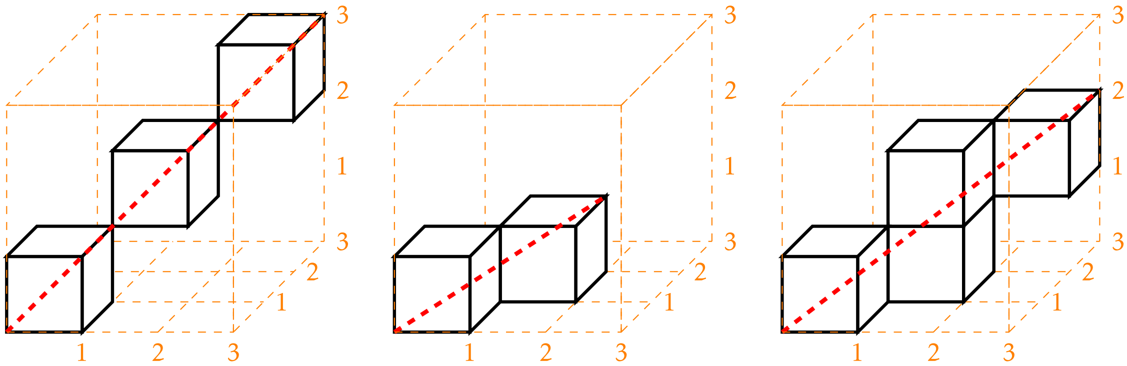
Hit the cubes

Time limit: 1s

You are at the origin $(0,0,0)$ with a very powerful laser gun. There is a slap of $n \times n \times n$ unit cubes in from of you. You want to shoot your gun to a point (x,y,z) with integer coordinates such that $0 \leq x,y,z \leq n$. Your laser gun is very precise and so the beam will stop exactly at point (x,y,z). What is the maximum number of cubes that you can hit?

Example

Suppose that $n = 3$. The following figure shows which cubes are hit when shooting at points $(3,3,3),(2,1,2)$ and $(3,2,3)$. The maximum that can achieved with $n = 3$ is 4 cubes.



Shooting at $(3,3,3)$ Shooting at $(2,1,2)$ Shooting at $(3,2,3)$

Input

A single integer n giving the size of the grid.

Constraints

- $1 \leq n \leq 10^8$

Output

A single line with the maximum number of cubes that the laser can traverse.

Sample test cases

Input 1	Output 1
3	4

Input 2	Output 2
8	18

Problem D
Farmer's dealema
Time limit: 2s

A farmer has n goods that he wants to transport from one side of a river to the other. To do so, he needs to build a boat and then perform some number of trips bringing goods from one side to the other.

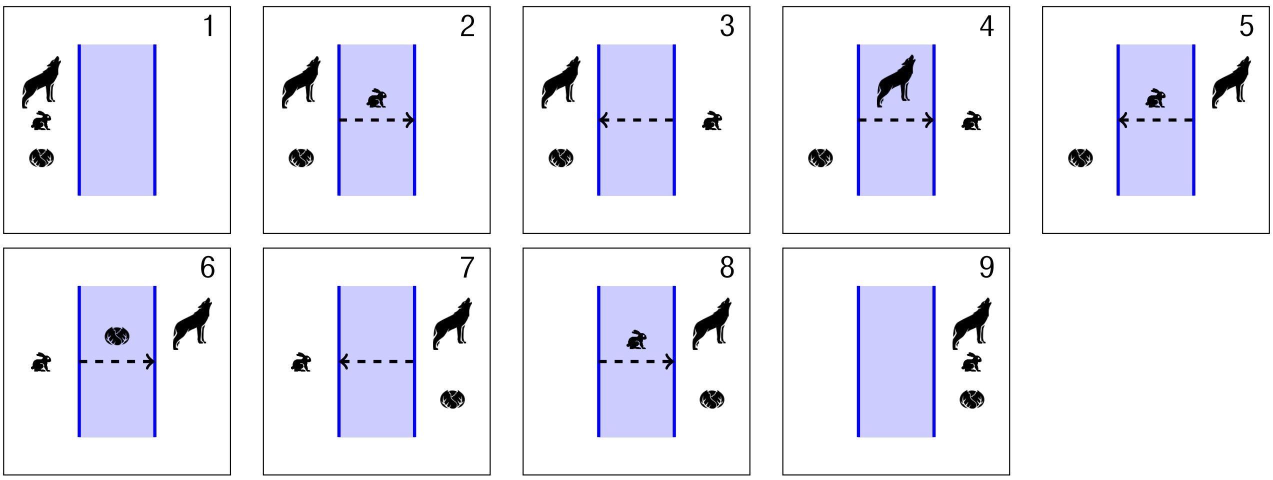
Unfortunately life is not that easy. Some goods cannot be left alone together without the farmer's supervision. Goods are said to be supervised by the farmer if the farmer is on the same side them. When the farmer is traversing the river, only the goods on the boat are under his supervision.

He wants to know what is the minimum boat capacity that he needs in order to safely transport all of his good across the river. Can you help him?

Example:

Suppose that $n = 3$ and the goods are: a wolf, a rabbit and a cabbage. The rabbit and the wolf cannot be left without supervision. The same goes for the rabbit and the cabbage.

The following figure shows how the farmer can safely transport them across the river with a boat of capacity 1 (the farmer is not counted in the capacity).



Input

The first line contains two integers n and m separated by a single space giving the number of goods and the number of pairwise restrictions between them.

Then follow m lines each with two integers x and y giving that goods x and y cannot be left without supervision.

For simplicity good are represented by numbers from 0 to $n-1$.

Constraints

- $1 \leq n \leq 20$
- $0 \leq m \leq n(n-1)/2$
- $0 \leq x, y < n$

Output

The first line of the output must contain a single integer k, the size of the boat (excluding the space for the farmer). The boat size you provide needs not to be optimal but has to satisfy $k_{min} \leq k \leq k_{min} + \log_2(k_{min})$ where k_{min} is the minimum boat capacity for which there exists a solution.

The second line must contain a single integer with the number of trips t in your solution. The number of trips can be any number up to n^2.

The follow t lines each with the list of passengers on the boat on each of the trips. The first trip is a trip from the left to the right, the second from right to left and so on.

You are allowed to perform trips without passengers. In those cases just print a blank line for the trip.

The farmer must be on the right side at the end.

Sample test cases

Input 1

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

Output 1

```
1
7
1

0
1
2

1
```

Input 2

```
4 0
```

Output 2

```
1
7
0

1

2

3
```

This page is not really blank.

Belgium Algorithm Contest

Problem E
Going the long way
Time limit: 2s

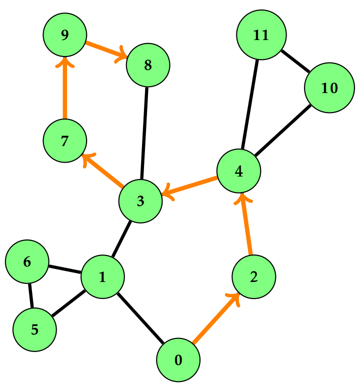
Let G be a graph. A path in G is a sequence of nodes (elements of $V(G)$) , $(v_1, v_2, \ldots, v_k)$ such that for all i, $(v_i, v_{i+1}) \in E(G)$ and for all i, j, $v_i \neq v_j$.

A cycle on G is a path $(v_1, v_2, \ldots, v_k)$ to which we append node v_1.

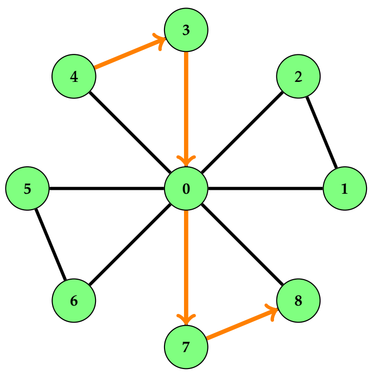
You will be given a connected graph G such that **any two distinct cycles have at most one node in common** and two nodes $s, t \in V$. Your task is to compute the length of the longest path from s to t in G. The graph is not weighted so the length of the path is defined as the number of edges in the path.

Example:

The following graphs represent the sample test cases.



$s = 0$ and $t = 8$



$s = 4$ and $t = 8$

Input

The first line contains two integers n and m giving the number of vertices and edges in the graph, respectively.

The follow m lines. The i-th line contains two integers v_i, u_i representing an undirected edge between nodes v_i and u_i in the graph.

No edge will be given more than once.

The last line contains two integers s and t as described above.

Constraints

- $2 \leq n \leq 10^5$
- $1 \leq m \leq \min(2 \cdot 10^5, n(n-1)/2)$
- $0 \leq v_i, u_i < n$
- $0 \leq s, t < n$

Output

A single line with the length of the longest path in G from s to t.

Be aware of your recursion depth if you program in Java to avoid RTE.

Sample test cases

Input 1

```
12 15
0 1
1 3
3 4
4 2
2 0
1 6
6 5
5 1
3 7
7 9
9 8
8 3
4 11
11 10
10 4
0 8
```

Output 1

```
6
```

Input 2

```
9 12
0 1
1 2
2 0
0 3
3 4
4 0
0 5
5 6
6 0
0 7
7 8
8 0
4 8
```

Output 2

```
4
```

This page is not really blank.

This page is not really blank.