Task: DLU
Long travels

XXVI OI, Stage III, Day two. Source file `dlu.*` Available memory: 128 MB. *11.04.2019*

On his numerous travels undertaken during the 25 years of the Polish Olympiad in Informatics, Byteasar has made many friends. On this anniversary, he would like to visit them all, which means traveling all over Byteotia again. This time around though, Byteasar plans to travel by air. There are n towns in Byteotia, each with its own airport. There are m direct bidirectional flight routes linking the cities, and the route network is designed so that it is possible to fly between any pair of towns, though this may involve transfers.

Known for his frugality, Byteasar wants to minimize the cost of his travel. As an experienced traveler, he had everything planned already, but then a new infrastructural bill has fixed the price of any direct domestic flight at one Bythaler. This changes everything, and now Byteasar struggles with p questions of the form "How much would it cost to fly from one friend who lives in town s_i to another who lives in town t_i?" Be a good sport and help him find the cheapest connections.

It stroke you that the pairs of friends who Byteasar inquires about live rather far apart — specifically, for each pair, every route connecting them involves at least $\frac{n}{10}$ flights. Answer Byteasar's queries so that he has a chance of visiting all his friends before the next edition of the Olympiad!

Input

The fist line of the standard input contains three integers n, m, and p ($2 \le n \le 100\,000$, $n-1 \le m \le 200\,000$, $1 \le p \le 200\,000$), separated by single spaces, which denote the numbers of towns, flight routes, and queries respectively. The towns are numbered from 1 to n.

The m lines that follow describe the flight routes; the i-th such line contains two integers a_i, b_i ($1 \le a_i, b_i \le n$, $a_i \ne b_i$), separated by a single space, which indicate that there is a bidirectional flight route linking towns a_i and b_i. Every flight route is described in at most one line.

The p lines that follow contain the queries; the i-th such line holds two integers s_i, t_i ($1 \le s_i, t_i \le n$, $s_i \ne t_i$) separated by a single space, which stand for a query for the price (i.e., number of flights along the shortest route) from town s_i to town t_i. It is known that each of these prices is at least $\frac{n}{10}$ bythalers.

Output

Your program should print p lines to the standard output; the i-th of these should contain a single integer: the answer to the i-th input query.

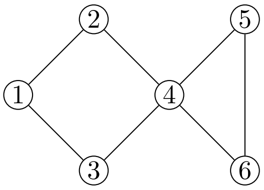
Example

For the input data:

```
6 7 2
1 2
2 4
3 1
3 4
4 5
4 6
6 5
2 5
1 6
```

the correct result is:

```
2
3
```



Sample grading tests:

`1ocen`: $n = 10$, $m = 30$, $p = 45$;

`2ocen`: $n = 100$, flight routes form a circle;

`3ocen`: $n = 100\,000$, flight routes form two circles which meet (intersect) in a single town.

Grading

The set of tests consists of the following subsets. Within each subset, there may be several tests.

Time limits for particular subsets are published in SIO.

Subset	Conditions	Score
1	$p = 1$	7
2	$m = n - 1$, each town has at most 2 (direct) flight routes	8
3	$m = n - 1$	9
4	$m = n$	16
5	town no. a, for $a \in \{1, \ldots, n\}$, may have flight routes only to towns no. $a - 5$, $a - 1$, $a + 1$, and $a + 5$	19
6	$p \leq 50\,000$	20
7	no further conditions	21