F. Koala and Notebook

time limit per test: 2 seconds
memory limit per test: 512 megabytes
input: standard input
output: standard output

Koala Land consists of m bidirectional roads connecting n cities. The roads are numbered from 1 to m by order in input. It is guaranteed, that one can reach any city from every other city.

Koala starts traveling from city 1. Whenever he travels on a road, he writes its number down in his notebook. He doesn't put spaces between the numbers, so they all get concatenated into a single number.

Before embarking on his trip, Koala is curious about the resulting number for all possible destinations. For each possible destination, what is the smallest number he could have written for it?

Since these numbers may be quite large, print their remainders modulo $10^9 + 7$. Please note, that you need to compute the remainder of the minimum possible number, **not** the minimum possible remainder.

Input

The first line contains two integers n and m ($2 \le n \le 10^5, n - 1 \le m \le 10^5$), the number of cities and the number of roads, respectively.

The i-th of the following m lines contains integers x_i and y_i ($1 \le x_i, y_i \le n, x_i \ne y_i$), representing a bidirectional road between cities x_i and y_i.

It is guaranteed, that for any pair of cities there is at most one road connecting them, and that one can reach any city from every other city.

Output

Print $n - 1$ integers, the answer for every city except for the first city.

The i-th integer should be equal to the smallest number he could have written for destination $i + 1$. Since this number may be large, output its remainder modulo $10^9 + 7$.

Examples

input Copy

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

output Copy

```
1
12
123
1234
12345
123456
1234567
12345678
123456789
345678826
```

input Copy

```
12 19
1 2
2 3
2 4
2 5
2 6
2 7
2 8
2 9
2 10
3 11
11 12
1 3
1 4
1 5
1 6
1 7
1 8
1 9
1 10
```

output Copy

```
1
12
13
14
15
16
17
18
19
1210
121011
```

input Copy

```
12 14
1 2
2 3
3 4
4 5
5 6
6 7
7 8
8 9
9 10
10 11
11 12
1 3
1 4
1 10
```

output Copy

```
1
12
13
134
1345
13456
1498
149
14
1410
141011
```

- Announcement (en)
- Tutorial (en)