

E. Lucky Country

time limit per test: 1 second

memory limit per test: 256 megabytes

input: standard input

output: standard output

Petya loves lucky numbers. Everybody knows that positive integers are lucky if their decimal representation doesn't contain digits other than 4 and 7. For example, numbers 47, 744, 4 are lucky and 5, 17, 467 are not.

One night Petya was sleeping. He was dreaming of being the president of some island country. The country is represented by islands connected by two-way roads. Between some islands there is no road way, even through other islands, that's why the country is divided into several regions. More formally, each island belongs to exactly one region, there is a path between any two islands located in the same region; there is no path between any two islands from different regions. A region is lucky if the amount of islands in it is a lucky number.

As a real president, Petya first decided to build a presidential palace. Being a lucky numbers' fan, Petya wants to position his palace in one of the lucky regions. However, it is possible that initially the country has no such regions. In this case Petya can build additional roads between different regions, thus joining them. Find the minimum number of roads needed to build to create a lucky region.

Input

The first line contains two integers n and m ($1 \leq n, m \leq 10^5$). They are the number of islands and the number of roads correspondingly. Next m lines contain road descriptions. Each road is defined by the numbers of islands that it connects: that is, by two integers u and v ($1 \leq u, v \leq n$). Some roads can connect an island with itself; there can be more than one road between a pair of islands. Numbers in each line are separated by exactly one space character.

Output

If there's no solution, output the only number "-1" (without the quotes). Otherwise, output the minimum number of roads r that need to be built to get a lucky region.

Examples

input
4 3 1 2 2 3 1 3
output
1

input
5 4 1 2 3 4 4 5 3 5
output
-1