

D. DZY Loves Games

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Today DZY begins to play an old game. In this game, he is in a big maze with n rooms connected by m corridors (each corridor allows to move in both directions). You can assume that all the rooms are connected with corridors directly or indirectly.

DZY has got lost in the maze. Currently he is in the first room and has k lives. He will act like the follows:

- Firstly he will randomly pick one of the corridors going from his current room. Each outgoing corridor has the same probability to be picked.
- Then he will go through the corridor and then the process repeats.

There are some rooms which have traps in them. The first room definitely has no trap, the n -th room definitely has a trap. Each time DZY enters one of these rooms, he will lost one life. Now, DZY knows that if he enters the n -th room with exactly 2 lives, firstly he will lost one live, but then he will open a bonus round. He wants to know the probability for him to open the bonus round. Please, help him.

Input

The first line contains three integers n, m, k ($2 \leq n \leq 500$; $1 \leq m \leq 10^5$; $2 \leq k \leq 10^9$).

The second line contains n integers, each of them is either 0 or 1. If the i -th number is 1, then the i -th room has a trap, otherwise it has not a trap. **Please note**, that the number of rooms with a trap is no more than 101. It is guaranteed that the first room has no trap, and the n -th room has a trap.

Then m lines follows. Each of them contains two integers u_i, v_i ($1 \leq u_i, v_i \leq n$; $u_i \neq v_i$), meaning that current corridor connects two rooms u_i and v_i . It is guaranteed that the corridor system is connected.

Output

Print the only real number — the probability for DZY to open the bonus round. The answer will be considered correct if its relative or absolute error doesn't exceed 10^{-4} .

Examples

input
5 5 3 0 0 1 0 1 1 2 2 3 3 4 4 5 1 2
output
0.25000000

input
3 2 2 0 1 1 1 2 2 3
output
-0.00000000

input
2 1 3 0 1 1 2
output
1.00000000