tropgard

Botanist Somhed regularly takes groups of students to one of Thailand's largest tropical gardens. The landscape of this garden is composed of **N** fountains (numbered **0**, **1**, …, **N-1**) and **M** trails. Each trail connects a different pair of distinct fountains, and can be traveled in either direction. There is at least one trail leaving each fountain. These trails feature beautiful botanical collections that Somhed would like to see. Each group can start their trip at any fountain.

Somhed loves beautiful tropical plants. Therefore, from any fountain he and his students will take the most beautiful trail leaving that fountain, *unless* it is the most recent trail taken and there is an alternative. In that case, they will take the second most beautiful trail instead. Of course, if there is no alternative, they will walk back, using the same trail for the second time. Note that since Somhed is a professional botanist, no two trails are considered equally beautiful for him.

His students are not very interested in the plants. However, they would love to have lunch at a premium restaurant located beside fountain number **P**. Somhed knows that his students will become hungry after taking exactly **K** trails, where **K** could be different for each group of students. Somhed wonders how many different routes he could choose for each group, given that:

- each group can start at any fountain;
- the successive trails must be chosen in the way described above; and
- each group must finish at fountain number **P** after traversing exactly **K** trails.

Note that they may pass fountain number **P** earlier on their route, although they still need to finish their route at fountain number **P**.

Your Task

Given the information on the fountains and the trails, you have to find the answers for **Q** groups of students; that is, **Q** values of **K**.

Write a program that takes the following parameters and counts the number of routes:

- **N** — the number of fountains. The fountains are numbered **0** through **N-1**.
- **M** — the number of trails. The trails are numbered **0** through **M-1**. The trails will be given in *decreasing* order of beauty: for **0 ≤ i < M-1**, trail **i** is more beautiful than trail **i+1**.
- **P** — the fountain at which the premium restaurant is located.
- **R** — a two-dimensional array representing the trails. For **0 ≤ i < M**, trail i connects the fountains **R[i][0]** and **R[i][1]**. Recall that each trail joins a pair of distinct fountains, and no two trails join the same pair of fountains.
- **Q** — the number of groups of students.
- **G** — a one-dimensional array of integers containing the values of **K**. For **0 ≤ i < Q**, **G[i]** is the number of trails **K** that the **i**-th group will take.

For **0 ≤ i < Q**, your program must find the number of possible routes with exactly **G[i]** trails that group **i** could possibly take to reach fountain **P**. For each group **i**, your program should output **X** to report that the number of routes is **X**. The answers must be given in the same order as the groups. If there are no valid routes, your program must output **0**.

Click here for an interactive demo!

Examples

Example 1

Consider the case shown in Figure 1, where **N=6**, **M=6**, **P=0**, **Q=1**, **G[0]=3**, and

$$R=\begin{array}{cc}1 & 2\\0 & 1\\0 & 3\\3 & 4\\4 & 5\\1 & 5\end{array}$$

Figure 1.

Note that trails are listed in decreasing order of beauty. That is, trail 0 is the most beautiful one, trail 1 is the second most beautiful one, and so on.

There are only two possible valid routes that follow 3 trails:

- 1 → 2 → 1 → 0, and
- 5 → 4 → 3 → 0.

The first route starts at fountain 1. The most beautiful trail from here leads to fountain 2. At fountain 2, the group has no choice, they must return using the same trail. Back at fountain 1, the group will now avoid trail 0 and choose trail 1 instead. This trail does indeed bring them to the fountain **P**=0.

Thus, the program should output **2**.

Example 2

Consider the case shown in Figure 2, where **N=5**, **M=5**, **P=2**, **Q=2**, **G[0]=3**, **G[1]=1**, and

$$R=\begin{array}{cc}1 & 0\\1 & 2\\3 & 2\\1 & 3\\4 & 2\end{array}$$

Figure

For the first group, there is only one valid route that reaches fountain 2 after following 3 trails: 1 → 0 → 1 → 2.

For the second group, there are two valid routes that reach fountain 2 after following 1 trail: 3 → 2, and 4 → 2.

Therefore, the correct program should first output **1** to report the answer for the first group, and then output **2** to report the answer for the second group.

Input Format

- Line 1: **N**, **M**, and **P**
- Lines 2 to **M**+1: description of the trails; i.e., line **i**+2 contains **R[i][0]** and **R[i][1]**, separated by a space, for **0 ≤ i < M**.
- Line **M**+2: **Q**.
- Line **M**+3: array **G** as a sequence of space-separated integers.

Output Format

One line containing **Q** space separated integers.

Sample Input 1	Sample Input 2
5 5 2 1 0 1 2 3 2 1 3 4 2 2 3 1	6 6 0 1 2 0 1 0 3 3 4 4 5 1 5 1 3
Sample Output 1	**Sample Output 2**
1 2	2

Subtasks

Subtask 1 (49% of points)

- 2 ≤ **N** ≤ 1 000
- 1 ≤ **M** ≤ 10 000
- **Q** = 1
- each element of **G** is an integer between 1 and 100, inclusive.

Subtask 2 (20% of points)

- 2 ≤ **N** ≤ 150 000
- 1 ≤ **M** ≤ 150 000
- **Q** = 1
- each element of **G** is an integer between 1 and 1 000 000 000, inclusive.

Subtask 3 (31% of points)

- 2 ≤ **N** ≤ 150 000
- 1 ≤ **M** ≤ 150 000
- 1 ≤ **Q** ≤ 2 000
- each element of **G** is an integer between 1 and 1 000 000 000, inclusive.