

## Тропический сад

Ботаник Сомхед часто организует экскурсии групп студентов в один из самых больших тропических садов Таиланда. Этот сад состоит из  $N$  фонтанов, пронумерованных числами от  $0$  до  $(N-1)$ , и  $M$  тропинок. Каждая тропинка соединяет пару различных фонтанов, и по ней можно ходить в обоих направлениях. При этом различные тропинки соединяют различные пары. От каждого фонтана ведет хотя бы одна тропинка. Вдоль каждой из тропинок находятся красивые коллекции растений, которые хотел бы увидеть Сомхед. Каждая группа может начать свою экскурсию от любого из фонтанов.

Сомхед любит тропические растения. Поэтому от каждого из фонтанов он и его группа студентов пойдут по наиболее красивой тропинке, которая ведет от этого фонтана, *за одним исключением*: если самой красивой является та тропинка, по которой они только что прошли. В этом случае, если есть другие тропинки, идущие от этого фонтана, то Сомхед и его группа пойдут по второй по красоте тропинке. Конечно, если другой тропинки нет, они вернутся по той же тропинке, по которой и пришли, используя эту тропинку второй раз подряд. Так как Сомхед — профессиональный ботаник, то среди тропинок нет двух одинаково красивых для него.

Студенты не очень интересуются растениями. Однако им бы хотелось пообедать в шикарном ресторане, который находится у фонтана с номером  $P$ . Сомхед знает, что его студенты проголодаются после того, как пройдут ровно  $K$  тропинок, где  $K$  может быть различным для разных групп студентов. Сомхеда интересует, сколько различных маршрутов он может выбрать для каждой из групп студентов, если:

- каждая группа может начинать прогулку от любого фонтана;
- последующая тропинка всегда выбирается описанным выше способом;
- каждая группа должна прийти к фонтану с номером  $P$ , пройдя ровно по  $K$  тропинкам.

Необходимо заметить, что по пути группа студентов может проходить мимо фонтана с номером  $P$ , следуя по своему маршруту, тем не менее, маршрут обязательно должен заканчиваться у фонтана с номером  $P$ .

### Задание

По данной информации о фонтанах и тропинках, необходимо найти количество различных маршрутов для каждой из  $Q$  групп студентов, то есть,  $Q$  значений  $K$ .

Необходимо написать процедуру `count_routes(N,M,P,R,Q,G)` которая принимает следующие параметры:

- $N$  — количество фонтанов. Фонтаны нумеруются от  $0$  до  $(N-1)$ .
- $M$  — количество тропинок. Тропинки нумеруются от  $0$  до  $(M-1)$ . Тропинки заданы в порядке убывания красоты: для  $0 \leq i < M-1$ , тропинка с номером  $i$  более красива, чем тропинка с номером  $(i+1)$ .
- $P$  — номер фонтана, возле которого находится шикарный ресторан.
- $R$  — двумерный массив, в котором описываются тропинки. Для  $0 \leq i < M$ , тропинка

с номером  $i$  соединяет фонтаны с номерами  $R[i][0]$  и  $R[i][1]$ . Следует учесть, что каждая тропинка соединяет различные фонтаны, и никакие две тропинки не соединяют одну и ту же пару фонтанов.

- $Q$  — количество групп студентов.
- $G$  — одномерный массив целых чисел, содержащий значения  $K$ . Для  $0 \leq i < Q$  в элементе массива  $G[i]$  задано количество тропинок  $K$ , по которым должна пройти  $i$ -ая группа.

Для  $0 \leq i < Q$  процедура должна найти количество возможных маршрутов с количеством тропинок, равным  $G[i]$ , по которым может пройти  $i$ -ая группа, чтобы попасть к фонтану с номером  $P$ . Для каждой группы с номером  $i$  процедура должна вызывать процедуру  $answer(X)$ , чтобы сообщить, что число маршрутов равно  $X$ . Ответы необходимо выдавать в том же порядке, в котором заданы группы. Если допустимых маршрутов не существует, процедура должна вызывать процедуру  $answer(0)$ .

## Примеры

### Пример 1

Рассмотрим пример, показанный на рис. 1, где  $N=6$ ,  $M=6$ ,  $P=0$ ,  $Q=1$ ,  $G[0]=3$  и

$R=$   
 1 2  
 0 1  
 0 3  
 3 4  
 4 5  
 1 5

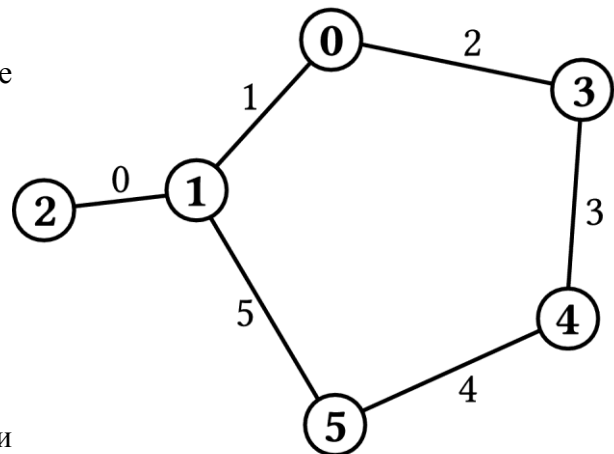


Рис. 1

Необходимо обратить внимание, что тропинки перечислены в порядке убывания красоты. Это значит, что тропинка с номером 0 является самой красивой, тропинка с номером 1 является второй по красоте и так далее.

Есть всего два допустимых маршрута, проходящих по трём тропинкам:

- $1 \rightarrow 2 \rightarrow 1 \rightarrow 0$ , и
- $5 \rightarrow 4 \rightarrow 3 \rightarrow 0$ .

Первый маршрут начинается возле фонтана с номером 1. Самая красивая тропинка отсюда ведет к фонтану с номером 2. У фонтана с номером 2 у группы нет выбора, они должны вернуться, используя ту же тропинку. Теперь, снова находясь у фонтана с номером 1, группа не может использовать тропинку с номером 0, и вместо этого выбирает тропинку с номером 1. Эта тропинка приводит их к фонтану с номером  $P=0$ .

Таким образом, нужно вызвать процедуру  $answer(2)$ .

## Пример 2

Рассмотрим пример, показанный на рис. 2, где  $N=5$ ,  $M=5$ ,  $P=2$ ,  $Q=2$ ,  $G[0]=3$ ,  $G[1]=1$ , и

1 0  
1 2  
 $R=3$  2  
1 3  
4 2

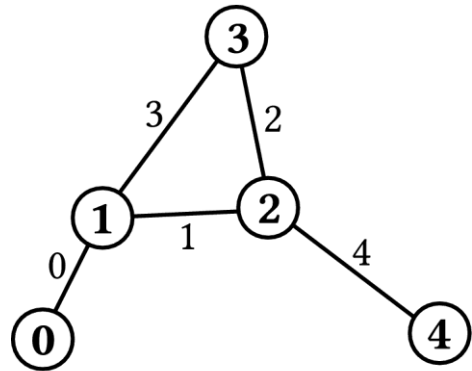


Рис. 2

Для первой группы есть только один маршрут, который приводит к фонтану с номером 2, проходя по трём тропинкам:  $1 \rightarrow 0 \rightarrow 1 \rightarrow 2$ .

Для второй группы есть два маршрута, которые приводят к фонтану с номером 2, проходя по одной тропинке:  $3 \rightarrow 2$ , и  $4 \rightarrow 2$ .

Таким образом, правильная реализация процедуры `count_routes` должна сначала вызывать `answer(1)` чтобы сообщить ответ для первой группы, после чего вызвать `answer(2)`, чтобы сообщить ответ для второй группы.

## Подзадачи

### Подзадача 1 (49 баллов)

- $2 \leq N \leq 1\,000$
- $1 \leq M \leq 10\,000$
- $Q = 1$
- Каждый элемент  $G$  является целым числом от 1 до 100 включительно.

### Подзадача 3 (31 балл)

- $2 \leq N \leq 150\,000$
- $1 \leq M \leq 150\,000$
- $1 \leq Q \leq 2\,000$
- Каждый элемент  $G$  является целым числом от 1 до 1 000 000 000 включительно.

### Подзадача 2 (20 баллов)

- $2 \leq N \leq 150\,000$
- $1 \leq M \leq 150\,000$
- $Q = 1$
- Каждый элемент  $G$  является целым числом от 1 до 1 000 000 000 включительно.

## Детали реализации

### Ограничения

- Ограничение по времени: 5 секунд
- Ограничение по памяти: 256 МВ  
**Замечание:** Нет отдельного ограничения на размер стека; используемая стеком память входит в общий объём используемой памяти.

### Интерфейс (API)

- Папка для разработки: garden/
- Участник должен разработать: garden.c или garden.cpp или garden.pas
- Интерфейс участника: garden.h или garden.pas
- Интерфейс модуля оценивания: gardenlib.h или gardenlib.pas
- Предлагаемый модуль оценивания: grader.c или grader.cpp или grader.pas
- Ввод для предлагаемого модуля оценивания: grader.in.1, grader.in.2, ...

**Замечание:** Предлагаемый модуль оценивания читает входной файл в следующем формате:

- Строка 1:  $N$ ,  $M$ , и  $P$ .
- Строки от 2 до  $(M+1)$ : описание тропинок, то есть, строка с номером  $(i+2)$  содержит для  $0 \leq i < M$  два числа  $R[i][0]$  и  $R[i][1]$ , разделенных пробелом.
- Строка  $(M+2)$ :  $Q$ .
- Строка  $(M+3)$ : массив  $G$ , как последовательность разделенных пробелами целых чисел.
- Строка  $(M+4)$ : массив ожидаемых ответов как последовательность разделенных пробелами целых чисел.
- Ожидаемый вывод для прилагаемого модуля оценивания: grader.expect.1, grader.expect.2, ... В этой задаче каждый из перечисленных файлов должен содержать только текст “**Correct.**”