

Задача А. Пляжный волейбол

Имя входного файла: volley.in
Имя выходного файла: volley.out
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Вера очень много работала в этом году, подавая своим коллегам пример настоящего труженика. На восьмое марта за прекрасное исполнение служебных обязанностей Вера получила подарок — долгожданный отпуск в Теплой Стране! Тяжелые трудовые будни закончились, и Вера уже нежится на пляже на берегу Теплового Моря.

Любимое хобби Веры — пляжный волейбол, и как же Вера ждала момента, когда она сможет испытать невероятный азарт этой игры! Вера уже познакомилась с несколькими симпатичными волейболистами, но она пока не решила, какая же команда достойна иметь в своем составе такого замечательного игрока.

Каждый из N капитанов команд мечтает заполучить Веру в состав своей команды, поэтому они хотят максимально проявить себя. Так как поиграть хотят все, они решили действовать следующим образом: все N команд выстроились в очередь. Первый матч играется между двумя командами, которые стоят в очереди раньше остальных. Победитель игры остается на площадке, а проигравший отправляется в конец очереди. В каждом из следующих матчей победитель предыдущего играет с первой командой из очереди, а проигравший в очередной встрече опять становится в конец очереди. Каждая команда имеет некоторую *силу*, причем для простоты будем предполагать, что силы команд различны, а победителем в матче является команда, сила которой больше. Матчей может быть как угодно много.

Вера решила для себя, что она будет действовать по самому справедливому принципу «считалочки»: она будет играть с одной из двух команд, играющих матч с соответствующим считалке номером K . Но затем Вера поняла, что уже выбрала себе команду, в которой хотела бы играть, причем ориентируясь не только на ее силу. Ей известны Q считалок, соответствующих различным значениям K . Для каждого из этих чисел K_i необходимо узнать, а кто же именно будет сражаться за столь ценный приз, то есть какие две команды будут играть в матче с номером K_i .

Формат входного файла

Первая строка входных данных содержит единственное целое число N — количество команд ($2 \leq N \leq 100\,000$). Вторая строка содержит N различных чисел от 1 до N — силы команд: первое число — сила команды, стоящей в начале очереди, второе — сила следующей по очереди команды, ..., последнее — сила команды, стоящей в конце очереди.

Третья строка содержит единственное целое число Q ($1 \leq Q \leq 100\,000$) — количество известных Вере считалок. Каждая из следующих Q строк содержит число K_i ($1 \leq K_i \leq 10^{18}$) — номер очередного интересующего Веру матча. Обратите внимание, K_i может быть больше N .

Формат выходного файла

Выведите Q строк: для каждого интересующего Веру числа K_i два числа в любом

порядке — **силы** команд, сыгравших на K_i -м шаге. Первая строка должна содержать ответ на первый запрос, вторая — на второй и так далее.

Примеры

volley.in	volley.out
4 1 3 2 4 1 3	3 4
4 2 1 4 3 3 1 5 2	2 1 4 2 2 4

Комментарии

Разберем первый тест из условия:

	Кто играет	Состояние очереди	Победитель	Проигравший
Матч №1	1 3	2 4	3	1
Матч №2	3 2	4 1	3	2
Матч №3	3 4	1 2	4	3

Таким образом, в единственном интересующем Веру третьем матче сыграют команды с силами 4 и 3.

Система оценки

Тесты к этой задаче состоят из четырех групп.

0. Тесты 1–2. Тесты из условия, оцениваются в ноль баллов.
1. Тесты 3–18. В тестах этой группы $N \leq 2\,000$, $Q = 1$, $K_i \leq 2\,000$. Эта группа оценивается в 30 баллов, баллы начисляются только при прохождении всех тестов группы.
2. Тесты 19–25. В тестах этой группы $N \leq 100\,000$, $1 \leq Q \leq 10$, $K_i \leq 100\,000$. Эта группа оценивается в 30 баллов, баллы начисляются только при прохождении всех тестов группы. Решение будет тестироваться на тестах этой группы только в случае прохождения всех тестов из первой группы.
3. В тестах этой группы дополнительные ограничения отсутствуют. Эта группа оценивается в 40 баллов. Решение будет тестироваться на тестах этой группы **offline**, т.е. после окончания тура, причем только в случае прохождения всех тестов из первой и второй групп. Тесты в этой группе оцениваются **независимо**.

Задача В. Турист Петр

Имя входного файла: `tourist.in`
Имя выходного файла: `tourist.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

На отдыхе в Теплой Стране Вера познакомилась с симпатичным волейболистом-трактористом Петром. Турист Петр, кстати, собирается после отличного отдыха в Теплой Стране отправиться в путешествие по городам Европы. Как известно, Европа обладает развитой транспортной системой: в Европе есть V интересующих Петра городов и E маршрутов ночных поездов. Каждый маршрут соединяет два различных города, время в пути составляет одну ночь. Поезда по маршруту ходят в обоих направлениях.

Основной целью поездки Петра является осмотр местных достопримечательностей. Поскольку Петр — невероятно занятой человек, то он решил, что все путешествие должно занимать не более четырех дней. Петр уже многое повидал, поэтому на осмотр достопримечательностей в каждом городе Петр тратит ровно один день. Он хочет составить наиболее практичный тур: каждый день он будет тратить на осмотр города, а каждую ночь — на переезд ночным поездом между городами. Разумеется, Петр не имеет ни малейшего желания посещать один город несколько раз.

Но на этом прагматичность Петра не заканчивается: Петр, как настоящий турист, хочет посмотреть на самые красивые европейские достопримечательности. Он долго изучал справочники и для каждого города оценил свою ожидаемую радость от его посещения p_i . Теперь он хочет найти маршрут, при котором его радость будет наибольшей. Помогите Петру найти такой маршрут.

Формат входного файла

В первой строке входных данных заданы два целых числа V и E ($1 \leq V, E \leq 3 \cdot 10^5$) — количество городов и маршрутов поездов, соответственно. В следующей строке заданы V целых чисел p_i ($1 \leq p_i \leq 10^8$), где p_i обозначает ожидаемую радость от посещения города с номером i . В следующих E строках заданы описания маршрутов поездов. Каждое описание состоит из пары различных чисел a_i и b_i ($1 \leq a_i, b_i \leq V$) — номеров городов, между которыми курсирует этот маршрут поезда. Гарантируется, что между каждой парой городов существует не более одного маршрута поезда.

Формат выходного файла

В первой строке выходных данных выведите число K ($1 \leq K \leq 4$) — количество городов в оптимальном маршруте туриста Петра. В следующей строке выведите номера этих городов в порядке посещения. Города нумеруются начиная с единицы. Если оптимальных маршрутов несколько, выведите любой из них.

Примеры

tourist.in	tourist.out
5 4 4 2 3 1 5 1 2 2 3 3 4 4 5	4 2 3 4 5
4 3 1 2 3 4 1 2 1 3 1 4	3 4 1 3

Система оценки

Тесты к этой задаче состоят из пяти групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов **предыдущих** групп.

- Тесты 1–2. Тесты из условия, оцениваются в ноль баллов.
- Тесты 3–16. В тестах этой группы $V, E \leq 100$. Эта группа оценивается в 20 баллов.
- Тесты 17–32. В тестах этой группы $V, E \leq 1000$. Эта группа оценивается в 20 баллов.
- Тесты 33–53. В тестах этой группы $V \leq 3000, E \leq 60000$. Эта группа оценивается в 30 баллов.
- В тестах этой группы дополнительные ограничения отсутствуют. Эта группа оценивается в 30 баллов. Решение будет тестироваться на тестах этой группы **offline**, т. е. после окончания тура.

Задача С. Шустрая черепашка

Имя входного файла: `turtle.in`
Имя выходного файла: `turtle.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Пафнутий и его друзья — большие любители разнообразных настольных игр. Особенно им нравятся игры, требующие как можно быстрее производить в уме непростые вычисления, поэтому абсолютным хитом их вечерних посиделок в аудиториях НУОП (Неизвестного университета олимпиадного программирования) стала игра «Шустрая черепашка». В комплект игры входят:

- Клетчатое поле из N рядов по M клеток. Каждая клетка поля либо свободна, либо блокирована для перемещения.
- Q игровых карточек. Каждая карточка содержит описание множества стартовых клеток A , множества дополнительно блокируемых клеток B и множества конечных

клеток C . Множества A , B и C непусты, попарно не пересекаются и состоят из свободных клеток.

- Маленькая фишка в форме черепашки.

Правила игры очень просты. Игроки последовательно разыгрывают игровые карточки. Как только открывается очередная карточка, игрокам необходимо вычислить, сколько существует *хороших* троек клеток (a_i, b_j, c_k) , где $a_i \in A$, $b_j \in B$, $c_k \in C$. Тройка клеток называется *хорошей*, если можно провести черепашку из стартовой клетки a_i в конечную клетку c_k , не посещая при этом клетку b_j . На перемещение черепашки наложено три условия:

1. Черепашка имеет право перемещаться только вниз и вправо в пределах поля.
2. Находиться на заблокированных клетках запрещено.
3. Клетка b_j также блокируется для перемещения.

Так как таблицу с правильными ответами создатели не включили в комплект, в пылу игры постоянно возникают споры о правильности того или иного значения. Для установления истины ребята попросили вас посчитать ответы для данного комплекта.

Формат входного файла

Первая строка входного файла содержит два целых числа N и M ($1 \leq N, M \leq 150$) — количество строк и столбцов игрового поля.

Следующие N строк по M символов описывают игровое поле в порядке следования сверху вниз, слева направо. Символ '.' соответствует свободной клетке, а '#' — занятой. Строки нумеруются от 1 до N , столбцы — от 1 до M .

Следующая строка содержит целое число Q ($1 \leq Q \leq 100\,000$) — количество игровых карточек.

Далее следуют Q блоков, описывающих карточки. Каждый блок состоит из трех строк, описывающих множества A , B и C соответственно. Первое число описания определяет размер соответствующего множества, после чего перечисляются его клетки. Каждая клетка задается двумя числами — номером строки и номером столбца. Все клетки в описании различны. Смотрите комментарии к примеру для лучшего понимания формата входных данных.

Гарантируется, что все множества непусты, все клетки всех множеств являются свободными и никакая клетка не принадлежит более чем одному множеству из какой-то карточки.

Гарантируется, что суммарный размер всех множеств на всех игровых карточках не превосходит 300 000, а именно: $\sum_{i=1}^Q (|A_i| + |B_i| + |C_i|) \leq 300\,000$.

Дополнительно гарантируется, что суммарное количество всех троек (и хороших, и нет) по всем карточкам не превосходит $2 \cdot 10^7$: $\sum_{i=1}^Q |A_i| \cdot |B_i| \cdot |C_i| = Q_{total} \leq 2 \cdot 10^7$.

Формат выходного файла

В выходной файл выведите ровно Q чисел по одному на строке — правильные ответы на карточки в порядке их следования во входном файле.

Пример

turtle.in	turtle.out
5 6	1
..##..	3
...#.	
..#.#.	
..#...#	
..#...	
2	
1 1 1	
3 2 1 2 3 4 3	
1 5 6	
2 1 2 2 1	
2 3 1 3 3	
2 5 1 5 6	

Комментарии

В приведенном примере игровой комплект содержит две карточки.

Во всех тройках первой карточки черепашка стартует в верхнем левом углу и финиширует в правом нижнем. Несложно видеть, что это возможно сделать, только если из трех элементов множества B блокируется первая клетка второй строки, то есть *хорошей* тройкой является $(1, 1) - (2, 1) - (5, 6)$.

На второй карточке *хорошими* являются тройки: $(1, 2) - (3, 1) - (5, 6)$, $(2, 1) - (3, 1) - (5, 6)$, $(2, 1) - (3, 1) - (5, 1)$.

Система оценки

Тесты к этой задаче состоят из четырех групп.

0. Тест 1. Тест из условия, оценивается в ноль баллов.

1. Тесты 2–18. В тестах этой группы $N \leq 100$, $Q_{total} \leq 1\,000$. Эта группа оценивается в 30 баллов. Баллы начисляются только при прохождении всех тестов группы.

2. Тесты 19–32. В тестах этой группы $N \leq 100$, $Q_{total} \leq 1\,000\,000$. Эта группа оценивается в 30 баллов. Баллы начисляются только при прохождении всех тестов группы. Решение будет тестироваться на тестах этой группы только в случае прохождения всех тестов из первой группы.

3. В тестах этой группы дополнительные ограничения отсутствуют, однако гарантируется, что N и Q_{total} будут равномерно возрастать с номером теста. Эта группа оценивается в 40 баллов. Решение будет тестироваться на тестах этой группы **offline**, т. е. после окончания тура, причем только в случае прохождения всех тестов из первой и второй групп. Тесты в этой группе оцениваются **независимо**.

Задача D. «Чапаев» на дереве

Имя входного файла: `game.in`
Имя выходного файла: `game.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

Вова и Марина любят играть в игры, а особенно — придумывать к ним свои правила. Недавно они открыли для себя веселую игру «Чапаев», в которой игроки должны сбивать щелчками шашки вражеского цвета с шахматной доски (также эта игра известна под названием «Щелкунчики»). Вдоволь наигравшись, они решили модифицировать правила, добавив игре математическую сложность.

Теперь они играют в «Чапаева» не на шахматной доске, а на доске в форме дерева! Их дерево состоит из N вершин. Вершина 1 является корнем дерева, а из каждой из оставшихся вершин проведено ребро в некоторую вершину с меньшим номером — ее непосредственного предка.

В игре участвуют шашки одного цвета, изначально расположенные в некоторых вершинах дерева. За один ход игрок выбирает некоторую шашку и щелчком отправляет ее к корню по ребрам дерева, сбивая при этом с доски все встреченные на пути шашки. Сама шашка, по которой производился удар, после попадания в корень дерева также слетает с доски.

Игроки делают ходы по очереди. Проигрывает тот игрок, к ходу которого на доске не остается шашек.

Придуманная ими игра замечательна также тем, что на одной и той же доске можно играть, начиная с разных начальных позиций шашек. Практика показала, что самые интересные партии получаются, если изначально расставить фишки во все вершины, являющиеся потомками (непосредственными или косвенными) некоторой вершины *Root*, при этом в саму вершину *Root* фишка не ставится.

Дети решили сыграть N партий, перебрав в качестве вершины *Root* каждую вершину дерева по одному разу. Если у очередной вершины *Root* нет потомков, и на доске изначально не оказывается ни одной фишки, то игры не происходит, и дети переходят к следующей расстановке. В каждой партии Марина ходит первой.

Вова интересуется у вас, в скольких партиях Марина сможет одержать победу, если игроки будут действовать оптимально.

Формат входного файла

В первой строке находится целое число N ($1 \leq N \leq 500\,000$) — количество вершин в дереве.

Во второй строке следуют целые числа p_2, p_3, \dots, p_N , разделенные пробелами, где p_i — это номер вершины, являющейся предком вершины i ($1 \leq p_i < i$).

Формат выходного файла

Выведите единственное целое число — количество партий, в которых Марина одержит победу.

Пример

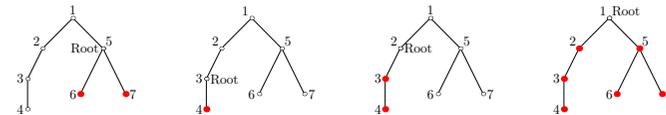
game.in	game.out
7 1 2 3 1 5 5	3

Комментарии

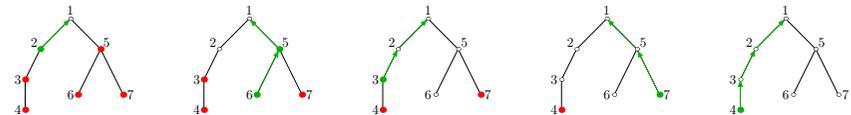
Разберем тест из условия. Доска для игры показана на рисунках ниже. Дети сыграют четыре партии, выбирая в качестве *Root* вершины 1, 2, 3 и 5. Если выбрать в качестве *Root* любую из трех оставшихся вершин, на доске исходно не окажется ни одной фишки, поэтому игры не произойдет.

Если выбрать в качестве *Root* вершину 5, фишки будут исходно находиться в вершинах 6 и 7. В такой партии Марина проигрывает: после того, как она сбивает любую из этих двух фишек с доски, Вова сбивает оставшуюся и заканчивает партию.

Если выбрать в качестве *Root* вершину 2 или 3, у Марины будет возможность выиграть игру за один ход, щелкнув по фишке из вершины 4 (при этом, в случае *Root* = 2, она по пути также собьет фишку из 3 вершины по правилам игры).



Можно убедиться, что если выбрать в качестве *Root* вершину 1, у Марины также будет выигрышная стратегия. Для этого первым ходом Марина должна сбить фишку из вершины 2. Пример партии с таким начальным расположением показан ниже.



Таким образом, Марина выигрывает в трех партиях.

Система оценки

Тесты к этой задаче состоят из пяти групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов **предыдущих** групп.

- Тест 1. Тест из условия, оценивается в ноль баллов.
- Тесты 2–17. В тестах этой группы $N \leq 20$. Эта группа оценивается в 20 баллов.
- Тесты 18–38. В тестах этой группы $N \leq 200$. Эта группа оценивается в 20 баллов.
- Тесты 39–59. В тестах этой группы $N \leq 5\,000$. Эта группа оценивается в 20 баллов.
- В тестах этой группы дополнительные ограничения отсутствуют. Эта группа оценивается в 40 баллов. Решение будет тестироваться на тестах этой группы **offline**, т. е. после окончания тура.