

Задача 1. Шоппинг

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Хейди находится в большом магазине. Ей нужно купить n предметов.

Сегодня ей повезло. Магазин проводит распродажу: за каждую покупку покупатель получает одну из скидок:

1. При покупке, содержащей не менее трех предметов, самый дешевый из них достаётся бесплатно.
2. При покупке, содержащей менее трех предметов, покупатель получает скидку $q\%$ на покупку.

Хейди должна купить все предметов из ее списка, каждый ровно в одном экземпляре. Она может сделать произвольное количество покупок. Для каждой из сделанных покупок применяется соответствующая скидка.

За какую минимальную суммарную стоимость она сможет купить все n предметов?

Формат входных данных

Первая строка содержит два целых числа n и q ($1 \leq n \leq 100\,000$, $0 \leq q \leq 100$) — число предметов в списке покупок Хейди и процент скидки, получаемой при покупке менее трех предметов.

Вторая строка содержит n целых чисел $p_1, p_2, \dots, p_n$ ($100 \leq p_i \leq 100\,000$) — стоимости предметов.

Гарантируется, что для всех $1 \leq i \leq n$ p_i делится на 100.

Формат выходных данных

Выведите одно целое число — минимальную суммарную стоимость покупки n предметов.

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Дополнительные ограничения	Необходимые подзадачи	Информация о проверке
0	0	Тесты из условия		полная
1	8	$n = 3$, $100 \leq p_i \leq 1\,000$		первая ошибка
2	18	$q = 0$		первая ошибка
3	16	$q = 40$		первая ошибка
4	22	$100 \leq p_i \leq 1\,000$	1	первая ошибка
5	36	нет	1 – 4	первая ошибка

Примеры

стандартный ввод	стандартный вывод
7 10 300 200 200 300 100 300 200	1090
3 20 1000 500 100	1280
4 0 200 100 300 200	600

Замечание

В первом примере три предмета ценой по 200 каждый могут быть куплены за одну покупку стоимостью 400 (один из предметов достанется бесплатно). Далее три предмета по 300 могут быть куплены за 600 (опять один бесплатно). Наконец, за покупку оставшегося предмета (ценой 100) мы получаем скидку 10%.

Во втором примере, если Хейди купит все три предмета за одну покупку, она получит скидку 100. Однако, если она купит каждый из предметов по отдельности, ее скидка составит $(1000 + 500 + 100) \cdot \frac{20}{100} = 320$.

Задача 2. Любвеобильная Луна

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

Луне пришла в голову дикая идея. Она выстроила своих $2n$ друзей в линию и дала каждому из них целое число от 1 до n , включительно. Каждое число было использовано ровно два раза. Каждые два друга, получившие одинаковое число, образуют пару.

Луна хочет отправить каждую из пар на свидание. Однако, это не так просто. Чтобы послать пару на свидание, два друга, образующих пару, должны стоять рядом, т.е. никто не должен стоять между ними.

Есть два возможных действия, которые Луна может предпринять:

- Она может поменять местами любых двух друзей, стоящих рядом.
- Если пара стоит рядом, Луна может послать её на свидание. Это убирает пару из линии. Оставшиеся друзья затем сдвигаются, чтобы заполнить образовавшуюся пустоту.

Действия можно осуществлять в любом порядке. Т.е. она может выполнить несколько обменов, затем послать несколько пар друзей на свидание, затем опять вернуться к обменам.

Найдите и выведите минимальное число действий, необходимое, чтобы всех послать на свидание.

Формат входных данных

Первая строка содержит единственное целое число n ($1 \leq n \leq 5 \cdot 10^5$).

Вторая строка содержит $2n$ целых чисел a_i ($1 \leq a_i \leq n$) — последовательность чисел, полученных друзьями при расстановке в линию, по порядку.

Формат выходных данных

Первая и единственная строка содержит минимальное число действий, необходимых выполнить Луне для отправки всех пар на свидание.

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Дополнительные ограничения	Необходимые подзадачи	Информация о проверке
0	0	Тесты из условия		полная
1	7	для каждой пары друзей нет никого другого, стоящего между ними, $n \leq 100$		первая ошибка
2	8	между друзьями в паре не более одного другого, $n \leq 100$	1	первая ошибка
3	11	первые n друзей получили числа от 1 до n , каждое ровно одно, не обязательно в таком порядке, $n \leq 3000$		первая ошибка
4	16	первые n друзей получили числа от 1 до n , каждое ровно одно, не обязательно в таком порядке	3	первая ошибка
5	22	$n \leq 3000$	1, 2, 3	первая ошибка
6	36	нет	1 — 5	первая ошибка

Примеры

стандартный ввод	стандартный вывод
3 3 1 2 1 2 3	4
5 5 1 2 3 2 3 1 4 5 4	7

Замечание

В первом примере, Луна должна начать с обмена третьего и четвертого друга. После этого обмена линия выглядит следующим образом: 3 1 1 2 2 3.

Затем, она должна послать пару с номером 1 и пару с номером 2 на свидание (в любом порядке). Как только она это сделает, два друга с номером 3 станут рядом и Луна сможет также послать их на свидание.

В итоге решение состоит из 4 действий: одного обмена и трёх свиданий.

Задача 3. Прыжки по отрезкам

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

Какое странное совпадение! Определив самую ценную коллекцию современного искусства, вы заметили, что она находится где-то недалеко от Липецка. Поскольку вы не знаете ее точного местоположения, вы хотите собрать больше информации. К счастью, в день вашего приезда, местное арт-сообщество проводит мероприятия, посвященные современному искусству. Похоже, это как раз та возможность, которую вы ждали.

Чтобы спланировать посещение этих мероприятий, вы пронумеровали их от 1 до n ; i -е мероприятие начинается в момент времени s_i и заканчивается в момент времени e_i .

Вы хотите начать свой визит с посещения какого-то события f и закончить визит на каком-то событии t . Пока вы не посетите событие t , вы всегда будете посещать ваше текущее мероприятие до конца, а затем сразу же переключитесь на другое мероприятие, которое проходит в данный момент. Это означает, что вы можете переключиться с события i на событие j тогда и только тогда, когда $s_j \leq e_i \leq e_j$.

Очевидно, что слишком частая смена событий заставит вас выглядеть подозрительно. Таким образом, вы хотите определить минимальное количество переключений событий, необходимых, если вы начнете с события f и закончите на t .

Кроме того, поскольку вы еще не знаете, когда вы прибудете в Липецк и когда вам нужно будет уехать вечером, вы хотите определить это для q различных пар начальных и конечных событий f и t .

Формат входных данных

Первая строка ввода содержит два целых числа n и q ($1 \leq n, q \leq 10^5$) — количество событий и количество запросов соответственно.

Затем следуют n строк, описывающих события.

В i -й из этих строк содержатся два целых числа s_i и e_i — время начала и окончания события i соответственно ($1 \leq s_i < e_i \leq 10^9$).

Затем следуют q строк, описывающих запросы.

В i -й из этих строк содержатся два целых числа f_i и t_i ($1 \leq f_i, t_i \leq n$). Это означает, что вы хотите определить минимальное количество переключений событий, необходимых для того, чтобы начать с события f_i и закончить визит на событии t_i .

Формат выходных данных

Выведите q строк. В i -й из этих строк выведите единственное целое число — минимальное количество переключений событий, необходимое для того, чтобы начать посещение с события f_i и завершить визит в событии t_i или строку «impossible», если этого добиться невозможно.

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Дополнительные ограничения	Необходимые подзадачи	Информация о проверке
0	0	Тесты из условия		полная
1	10	с каждого события можно переключиться не более чем на одно другое событие		первая ошибка
2	10	$n \leq 1000, q \leq 100$		первая ошибка
3	15	$n \leq 5000$		первая ошибка
4	15	$q \leq 100$	2	первая ошибка
5	20	никакое событие полностью не содержится в другом, иными словами не существует $i \neq j$, таких что $s_i \leq s_j < e_j \leq t_i$		первая ошибка
6	30	нет	1, 2, 3, 4, 5	первая ошибка

Примеры

стандартный ввод	стандартный вывод
5 2 1 3 2 4 4 7 7 9 3 7 1 4 3 2	2 impossible
8 5 1 2 3 4 1 5 6 7 5 10 10 20 15 20 999999999 1000000000 1 6 1 7 2 4 3 3 5 8	3 4 impossible 0 impossible

Замечание

В первом примере можно начать с события 1 и закончить событием 4, переключившись с события 1 на событие 5, а затем на событие 4, в результате чего произойдет два переключения событий.

Однако нет возможности начать на событии 3 и закончить на событии 2, потому что событие 2 заканчивается раньше, чем событие 3.

Задача 4. Опасные коровы

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	6 секунд
Ограничение по памяти:	256 мегабайт

Недавно коровы начали заболеть Очень Опасным Вирусом Коров (ООВК), который делает коров опасными для туристов. После нескольких инцидентов с нападениями решено было отделить территории в Альпах, где пасутся коровы, от территорий, где гуляют туристы.

Вам дана карта Альп. На карте есть n территорий. На каждой из них могут либо пастись коровы, либо гулять туристы, либо она может быть свободной. Некоторые пары территорий соединены двусторонними тропинками. Каждая тропинка имеет некоторую неотрицательную длину. (Говоря терминами теории графов, карта представляет собой неориентированный взвешенный граф.)

Вы можете построить стены на некоторых территориях. Если на территории построена стена, то коровы и туристы не могут проходить через эту территорию.

Ваша задача – выбрать, на каких территориях построить стены. Это множество территорий должно удовлетворять следующим условиям:

- Оно должно состоять только из свободных территорий.
- Оно должно отделять территории, где пасутся коровы от территорий, где гуляют туристы. А именно, коровы не должны иметь возможность добраться по тропинкам от территории, на которой они пасутся, до территории, на которой гуляют туристы (не проходя по территории, на которой построена стена).
- Оно **не должно** разделять территории, на которых гуляют туристы. А именно, турист должен быть способен пройти по тропинкам от любой территории, где гуляют туристы, до любой другой территории, где гуляют туристы (не проходя через территорию, на которой построена стена).

Если существует несколько способов добиться описанного выше, то ваша цель — постараться сделать обслуживание стен проще. Стены будут обслуживать специально обученные техники. На каждой территории, где гуляют туристы, будет находиться специально обученный техник.

Для каждой свободной территории A определим ее *удаленность* как минимальную длину пути по тропинкам от A до какой-либо территории, где гуляют туристы. (Длина пути равна сумме длин его тропинок. Обратите внимание, что пути, по которым будут ходить техники, **могут** проходить через территории со стеной и территории, где пасутся коровы, у техников будет все необходимое оборудование, чтобы проходить по таким территориям.)

Определим *удаленность* множества территорий как **максимальную** удаленность территории в этом множестве.

Из всех возможных наборов территорий, на которых будут построены стены, удовлетворяющие описанным условиям, выберите ту, которая имеет **минимальную возможную** удаленность. Если таких наборов несколько, выведите любой из них.

Обратите внимание, что количество территорий в наборе не имеет значения. В частности, **не требуется** строить как можно меньше стен.

Формат входных данных

Первая строка ввода содержит два целых числа n и m ($2 \leq n \leq 3 \cdot 10^5$, $n - 1 \leq m \leq 3 \cdot 10^5$) — количество территорий и тропинок, соответственно. Территории пронумерованы от 1 до n .

Вторая строка содержит n целых чисел $t_1, t_2, \dots, t_n$, где t_i равно -1 , если на i -й территории пасутся коровы, 0 , если она свободна, и 1 , если на ней гуляют туристы.

Следующие m строк описывают тропинки, j -я из них содержит три целых числа a_j , b_j и l_j ($1 \leq a_j < b_j \leq n$, $0 \leq l_j \leq 10^9$), эта тройка задает тропинку между территориями a_j и b_j длиной l_j .

Гарантируется, что:

- между любыми двумя территориями не более одной тропинки,
- можно пройти между любыми двумя территориями, используя тропинки,
- есть хотя бы одна территория, где пасутся коровы,
- есть хотя бы одна территория, где гуляют туристы.

Формат выходных данных

Если построить стены так, чтобы удовлетворить описанные условия, невозможно, выведите -1 .

В противном случае, в первой строке выведите целое число k — количество стен, которые вы хотите построить. Вторая строка должна содержать k целых чисел — номера территорий, на которых следует построить стены. (Это должны быть различные числа от 1 до n , включительно. Их можно выводить в любом порядке.)

Ваш ответ будет принят, если он представляет собой один из корректных ответов с минимальной удаленностью.

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Дополнительные ограничения	Необходимые подзадачи	Информация о проверке
0	0	Тесты из условия		полная
1	7	$n \leq 10$		первая ошибка
2	22	$l_j = 0$		первая ошибка
3	16	есть ровно одна территория, на которой гуляют туристы		первая ошибка
4	11	есть ровно $n - 1$ тропинка (граф представляет собой дерево)		первая ошибка
5	8	$n, m \leq 2000, l_j = 1$		первая ошибка
6	36	нет	1, 2, 3, 4, 5	первая ошибка

Примеры

стандартный ввод	стандартный вывод
10 14 1 0 1 0 0 0 0 0 -1 -1 1 2 1 1 6 1 2 3 1 2 5 2 3 4 1 4 5 1 4 8 2 5 6 1 5 7 1 6 7 2 6 10 1 7 8 1 7 9 1 8 9 1	3 4 5 6
5 5 1 0 0 -1 0 1 2 1000 2 3 1000 3 4 10 4 5 10 1 5 10	2 3 5
4 3 1 0 -1 1 1 2 0 2 3 21 2 4 13	-1

Замечание

В первом примере минимальная возможная удаленность равна 2, она может быть получена строительством стен на территориях 4, 5, и 6. Обратите внимание, что нельзя построить стены на территориях 4, 2 и 6, даже несмотря на то, что в результате удаленность будет равна 1, так как если построить эти стены, будет невозможно переходить между территориями 1 и 3, на которых гуляют туристы, не проходя через стену.

Во втором примере удаленность территории 2 равна 1000, а удаленность территории 3 равна 30, ведь до нее можно добраться по пути 1–5–4–3. (Помните, специально обученные техники могут ходить по территориям, где построены стены или где пасутся коровы.) Следовательно стены оптимально построить на территориях 5 и 3 (но не 2), удаленность будет равна 30.