

Задача 5. Автоматизированное управление доставкой

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

Группа программистов регионального сортировочного центра работает над автоматизацией управления доставкой почты.

Посылки принимаются в клиентских почтовых пунктах. Почтовый пункт принимает посылки, вес каждой из которых составляет целое число килограммов. Минимальный вес посылки равен 1 кг, а максимальный вес — k кг. Принятые посылки помещаются в специальный пакет.

Если после приема очередной посылки суммарный вес посылок в пакете больше или равен x кг, то пакет доставляется в муниципальный почтовый центр, где пакет с посылками перемещается в специальный контейнер.

Если после доставки очередного пакета суммарный вес посылок в контейнере больше или равен y кг, то контейнер перевозится в региональный сортировочный центр, откуда посылки уже доставляются получателям.

Суммарный вес посылок в контейнере при его перевозке может различаться в зависимости от массы принятых посылок. Необходимо выяснить, каким может быть минимальный суммарный вес посылок в контейнере при перевозке его из муниципального почтового центра в региональный сортировочный центр.

Требуется написать программу, которая по заданным значениям k — максимального веса посылки, x — необходимого веса пакета для его отправки в муниципальный почтовый центр, и y — необходимого веса контейнера для его отправки в региональный сортировочный центр, определяет минимальный вес контейнера при его перевозке.

Формат входного файла

Входной файл содержит три целых положительных числа, по одному на строке. Первая строка содержит число k ($1 \leq k \leq 10^9$). Вторая строка содержит число x ($1 \leq x \leq 10^9$). Третья строка содержит число y ($1 \leq y \leq 10^9$).

Формат выходного файла

Требуется вывести одно целое число — минимальный возможный вес контейнера при перевозке.

Пример входных и выходных файлов

<code>delivery.in</code>	<code>delivery.out</code>
2 7 20	21

Пояснение к примеру

В приведенном примере принимаются посылки весом 1 и 2 кг. При накоплении посылок с суммарным весом хотя бы в 7 кг пакет доставляется из клиентского почтового пункта в муниципальный почтовый центр. При накоплении посылок с суммарным весом хотя бы в 20 кг контейнер перевозится из муниципального почтового центра в региональный сортировочный центр.

Минимальный возможный вес контейнера в данном примере составляет 21 кг и достигается, например, следующим образом: в муниципальный почтовый центр

последовательно доставляется 3 пакета по 7 кг каждый. Пакет весом 7 кг может получиться, например, после приема семи посылок по 1 кг.

Описание подзадач и системы оценивания

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

Подзадача	Баллы	Ограничения		Необходимые подзадачи
		k	x, y	
1	21	$k = 1$	$1 \leq x, y \leq 100$	
2	18	$k = 2$	$1 \leq x, y \leq 100$	
3	21	$1 \leq k \leq 100$	$1 \leq x, y \leq 100$	1, 2
4	17	$1 \leq k \leq 40\,000$	$1 \leq x, y \leq 40\,000$	1, 2, 3
5	23	$1 \leq k \leq 10^9$	$1 \leq x, y \leq 10^9$	1, 2, 3, 4

Получение информации о результатах окончательной проверки

По запросу сообщается результат окончательной проверки на каждом тесте.

Задача 6. Большой линейный коллайдер

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

Группа ученых работает в международной научной лаборатории, которая занимается исследованиями поведения элементарных частиц в установке для экспериментов «Большой линейный коллайдер» (БЛК). Установка БЛК представляет собой прямую, в некоторых точках которой размещаются частицы, которые могут перемещаться вдоль прямой.

В очередном эксперименте в БЛК размещаются n частиц, каждая из которых представляет собой либо отрицательно заряженную частицу — электрон e^- , либо положительно заряженную частицу — позитрон e^+ . В эксперименте i -я частица исходно размещается в точке с координатой x_i . После начала эксперимента в результате работы БЛК частицы начнут перемещаться в разные стороны вдоль прямой: e^- частицы перемещаются по направлению уменьшения координаты, а e^+ частицы — по направлению увеличения координаты. Абсолютные величины скоростей всех частиц одинаковы и равны 1.

Если в процессе перемещения частицы e^- и e^+ оказываются в одной точке, то они взаимодействуют и обе исчезают, при этом они не влияют на дальнейшее поведение остальных частиц.

Ученые выбрали m различных моментов времени t_1, t_2, \dots, t_m , для каждого из которых их интересует, какое количество частиц находится в БЛК непосредственно после каждого из этих моментов времени. Отсчет времени начинается с момента 0, когда частицы приходят в движение. Частицы, исчезнувшие в результате взаимодействия в момент времени t_j , не должны учитываться при подсчете количества частиц для этого момента времени.

Требуется написать программу, которая по описанию исходного расположения и типов частиц, а также заданным моментам времени, определяет для каждого из моментов количество частиц, которое будет находиться в БЛК непосредственно после этого момента.

Формат входного файла

Первая строка входного файла содержит число n — количество частиц ($1 \leq n \leq 200\,000$). Последующие n строк описывают частицы следующим образом: каждая строка содержит по два целых числа x_i и v_i — координату i -й частицы и ее тип соответственно ($-10^9 \leq x_1 < x_2 < \dots < x_n \leq 10^9$, v_i равно -1 или 1). Частица e^- описывается значением $v_i = -1$, а частица e^+ описывается значением $v_i = 1$.

Следующая строка содержит целое число m — количество моментов времени, которые выбрали ученые ($1 \leq m \leq 200\,000$). Последняя строка содержит m целых чисел: t_1, t_2, \dots, t_m ($0 \leq t_1 < t_2 < \dots < t_m \leq 10^9$).

Формат выходного файла

Для каждого момента времени во входном файле требуется вывести одно число: количество частиц в БЛК непосредственно после этого момента.

Примеры входных и выходных файлов

<code>linear.in</code>	<code>linear.out</code>
4 -1 1 0 -1 1 1 5 -1 4 0 1 2 3	4 2 0 0

Пояснение к примеру

В приведенном примере в начальный момент в БЛК находятся 4 частицы: частица e^+ в точке -1 , частица e^- в точке 0 , частица e^+ в точке 1 и частица e^- в точке 5 .

В момент времени 0.5 первая частица e^+ и первая частица e^- сталкиваются в точке с координатой -0.5 и исчезают. В момент времени 1 оставшиеся две частицы находятся в точках с координатами 2 и 4 , соответственно. В момент времени 2 они сталкиваются в точке 3 и исчезают. Больше в БЛК частиц нет.

Описание подзадач и системы оценивания

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

Под-задача	Баллы	Дополнительные ограничения				Необх. подзадачи
		n	x_i	m	t_i	
1	35	$1 \leq n \leq 100$	$-100 \leq x_i \leq 100$	$m = 1$	$0 \leq t_1 \leq 100$	
2	12	$1 \leq n \leq 100$	$-10^9 \leq x_i \leq 10^9$	$m = 1$	$0 \leq t_1 \leq 10^9$	1
3	12	$1 \leq n \leq 200\,000$	$-10^9 \leq x_i \leq 10^9$	$m = 1$	$0 \leq t_1 \leq 10^9$	1, 2
4	41	$1 \leq n \leq 200\,000$	$-10^9 \leq x_i \leq 10^9$	$1 \leq m \leq 200\,000$	$0 \leq t_1 \leq 10^9$	1, 2, 3

Получение информации о результатах окончательной проверки

По запросу сообщается результат окончательной проверки на каждом тесте.

Задача 7. Силовые поля

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

В физико-биологической лаборатории исследуют воздействие излучения на растения при облучении через *силовые поля*.

Экспериментальная установка содержит квадратную платформу размером $10^9 \times 10^9$, заполненную плодородной почвой. Над платформой установлен источник излучения. Между источником излучения и платформой можно включать n силовых полей.

Генератор силовых полей установлен над точкой $(0, 0)$. При этом i -е силовое поле представляет собой прямоугольник со сторонами, параллельными границам платформы и координатами двух противоположных углов $(0, 0)$ и (x_i, y_i) .

В эксперименте планируется изучать воздействие излучения на растения при облучении через k силовых полей. Из заданных n полей необходимо выбрать k полей для эксперимента. Ученые хотят выбрать силовые поля таким образом, чтобы площадь участка платформы, над которой находятся все k выбранных силовых полей, была максимальна.

Требуется написать программу, которая по заданным целым числам n , k и описанию n силовых полей определяет, какие k силовых полей необходимо выбрать для эксперимента, чтобы площадь участка, покрытого всеми k силовыми полями, была максимальна, и выводит площадь этого участка.

Формат входного файла

Первая строка входного файла содержит целые числа n и k ($1 \leq k \leq n \leq 200\,000$) — общее количество силовых полей и количество силовых полей, которые необходимо выбрать для эксперимента.

Последующие n строк содержат по два целых числа x_i , y_i ($1 \leq x_i, y_i \leq 10^9$) — координаты дальнего от начала координат угла прямоугольного участка i -го силового поля.

Формат выходного файла

Требуется вывести одно целое число: максимальную площадь искомого участка

Пример входных и выходных файлов

power.in	power.out
5 3 3 5 2 2 2 5 4 4 5 3	9

Пояснение к примеру

На рис. 1 показаны пять силовых полей, заданных во входном файле. Оптимальный способ выбрать из них три поля для эксперимента показан на рис. 2.

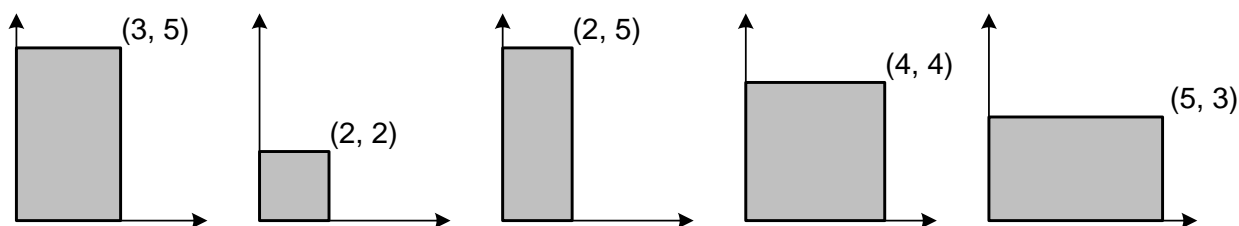


Рис 1. Силовые поля в примере описания входных данных.

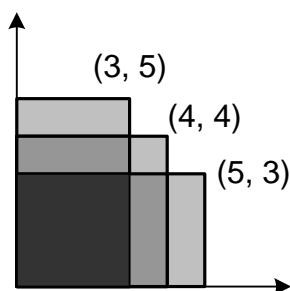


Рис 2. Оптимальный выбор трех из пяти силовых полей в данном примере.

Описание подзадач и системы оценивания

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

Подзадача	Баллы	Ограничения		Необходимые подзадачи
		n	k	
1	18	$1 \leq n \leq 20$	$1 \leq k \leq n$	
2	25	$1 \leq n \leq 300$	$1 \leq k \leq n$	1
3	20	$1 \leq n \leq 3\,000$	$1 \leq k \leq n$	1, 2
4	17	$2 \leq n \leq 200\,000$	$k = 2$	
5	20	$1 \leq n \leq 200\,000$	$1 \leq k \leq n$	1, 2, 3, 4

Получение информации о результатах окончательной проверки

По запросу сообщается результат окончательной проверки на каждом тесте.

Задача 8. Повышение квалификации

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

Взаимодействие сотрудников в некоторой компании организовано в виде иерархической структуры. Всего в компании работают n сотрудников. Каждому сотруднику присвоен уникальный номер от 1 до n , директору присвоен номер 1. У каждого сотрудника, кроме директора, есть ровно один непосредственный начальник. Непосредственный начальник сотрудника i имеет номер p_i , причем $p_i < i$.

Сотрудник x является подчиненным уровня 1 сотрудника y , если $p_x = y$. Для $k > 1$ сотрудник x является подчиненным уровня k сотрудника y , если сотрудник p_x является подчиненным уровня $k - 1$ сотрудника y .

У директора компании появилась возможность направить некоторых сотрудников на курсы повышения квалификации. Для этого он решил выбрать два числа L и R и направить на курсы всех сотрудников с номерами i , такими что $L \leq i \leq R$.

Перед тем, как выбрать числа L и R , директор получил m пожеланий от сотрудников компании, j -е пожелание задается двумя числами u_j и k_j и означает, что сотрудник u_j просит отправить на курсы одного из своих подчиненных уровня k_j . Для экономии средств директор хочет выбрать такие L и R , чтобы количество сотрудников, направленных на повышение квалификации, было минимальным возможным, но при этом все пожелания были выполнены.

Требуется написать программу, которая по заданным в компании отношениям начальник-подчиненный и пожеланиям сотрудников определяет такие числа L и R , что если отправить на курсы повышения квалификации всех сотрудников с номерами от L до R включительно, то все пожелания будут выполнены, а количество сотрудников, направленных на повышение квалификации, будет минимальным возможным. Если оптимальных пар чисел L, R будет несколько, требуется найти ту из них, в которой значение L минимально.

Формат входного файла

Первая строка входного файла содержит число n — количество сотрудников компании ($2 \leq n \leq 200\,000$). Вторая строка содержит $(n - 1)$ чисел: p_2, p_3, \dots, p_n ($1 \leq p_i < i$) — номера непосредственных начальников сотрудников.

Третья строка содержит число m — количество пожеланий от сотрудников.

Последующие m строк задают пожелания сотрудников и содержат по два целых числа u_j, k_j ($1 \leq u_j < n, 1 \leq k_j < n$, гарантируется, что у сотрудника u_j есть хотя бы один подчиненный уровня k_j).

Формат выходного файла

Необходимо вывести два искоемых числа: L и R . Если оптимальных пар (L, R) несколько, требуется вывести ту, в которой значение L минимально.

Пример входных и выходных файлов

<code>qual.in</code>	<code>qual.out</code>
7 1 1 2 2 3 3 3 1 1 3 1 1 2	3 6

Пояснение к примеру

На повышение квалификации будут направлены сотрудники с номерами 3, 4, 5 и 6. Сотрудник с номером 3 является подчиненным уровня 1 сотрудника с номером 1, сотрудник с номером 4 — подчиненным уровня 2 сотрудника с номером 1, а сотрудник с номером 6 — подчиненным уровня 1 сотрудника с номером 3.

Описание подзадач и системы оценивания

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

Подзадача	Баллы	Ограничения		Необходимые подзадачи
		n	Дополнительные условия	
1	19	$2 \leq n \leq 50$		
2	25	$2 \leq n \leq 3000$		1
3	21	$2 \leq n \leq 200\,000$	для всех i выполнено $p_i = i - 1$	
4	35	$2 \leq n \leq 200\,000$		1, 2, 3

Получение информации о результатах окончательной проверки

По запросу сообщаются баллы за каждую подзадачу.