Региональный этап, второй тур, 6 февраля 2017 г.

## Задача 5. Автоматизированное управление доставкой

ИМявходногофаЗла: delivery.in

ИмявыходногофаЙла: delivery.out Ограничениеповремени: lсекунда Ограничениепопамхти: 2Збмегабайт

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 х 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 *у* 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, х — необходимого веса пакета для его отправки в муниципальный почтовый центр, и у — 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 < /г < 109). Вторая строка содержит число х (1 < х < 109). Третья строка содержит число у (1 *у q<* 109).

#### *Формат выходного файла*

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

*Пример входных и выходных файлов*

|  |  |
| --- | --- |
| **delivery.in** | **delivery.out** |
| 2  7  20 | 21 |

Поясиеиие 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

GО]ЭТИ]ЭОВОЧНЫЙ ЦeHT]Э.

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

##### Региональный этап, второй тур, 6 февраля 2017 г.



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* | *•У* |
| 1 | 21 | *k ——* 1 | I *х, у q<* 100 |  |
| 2 | 18 | *k ——* 2 | I *х, у q<* 100 |  |
| 3 | 21 | 1 < /г < 100 | I *< х, у <* 100 | 1, 2 |
| 4 | 17 | 1 < /г < 40 000 | I *< х, у <* 40 000 | 1, 2, 3 |
| 5 | 23 | 1 < /г < 109 | 1 < , < 109 | 1, 2, 3, 4 |

*Получение информации о результатах окончательной проверки*

По запросу сообщается результат окончательной проверки на каждом тесте.

## Задача 6. Большой линейный коллайдер

ИМявходногофаЗла: linear.in

ИмявыходногофаЙла: linear.out Ограничениеповремени: lсекунда Ограничениепопамхти: 2Збмегабайт

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

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

Если в процессе перемещения частицы *e—* и *е!* оказываются в одной точке, то они взаимодействуют и обе исчезают, при этом они не влияют на дальнейшее поведение остальных частиц.

Ученые выбрали ш различных моментов времени h, t2, .. ., /<, ДЛЯ КІІЖДОГО ИЗ КОТО]ЭЫХ 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;, 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 п — количество частиц (1 < п < 200 000). Последующие в строк описывают частицы следующим образом: каждая строка содержит по два целых числа *х;* и г; — координату *i-й* частицы и ее тип соответственно (—109 < zi < 32 < . <x < 109, *г/* раВно —1 или 1). Частица *е* описывается значением *;* —1, а частица e+ описывается значением *v;* = 1.

Следующая строка содержит целое число m — количество моментов времени, которые выбрали ученые (1 < *т <* 200 000). Последняя строка содержит *т* целых чисел: f1, f2, .. ., /< (0 < 31 < /2 < . < / < 10’).

Формат выхоdлоао *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*

|  |  |  |  |
| --- | --- | --- | --- |
| **linear.in** | | | **linear.out** |
| 4 |  |  | 4 |
| 0  1  5 | -1  1  -1 |  | 2  0  0 |
| 4 |  |  |  |
| 0 | 1 2 | 3 |  |

Лояслелье х лрьмеру

В приведенном примере в начальный момент в БЛК находятся 4 частицы: частица *е+*

в точке —1, частица *е* в точке 0, частица *e+* в точке 1 и частица *е* в точке 5.

В момент времени 0.5 первая частица *e+ п* первая частица *е* 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.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Пол- задача | **Баллы** | **Дополнительные ограничения** | | | | **Необх.**   * одзадачи |
| П | А | Ш | *t* |
| 1 | 35 | 1 п 100 | —100 х; 100 | m = 1 | 0 ii 100 |  |
| 2 | 12 | 1 п 100 | —109 *х;* 10’ | m = 1 | 0 ii 10’ | 1 |
| 3 | 12 | 1 < п < 200 000 | —109 < х, < 109 | ш = 1 | 0 < /і < 109 | 1, 2 |
| 4 | 41 | 1 < п < 200 000 | —109 < *х;* < 109 | 1 < m < 200 000 | 0 < /і < 109 | 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 109 Х 109, заполненную плодородной почвой. Над платформой установлен источник излучения. Между источником излучения и платформой можно включать п силовых полей.

Генератор силовых полей установлен над точкой (0, 0). При этом *i-e* силовое поле представляет собой прямоугольник со сторонами, параллельными границам платформы и координатами двух противоположных углов (0, 0) и (х;, *yi).*

В эксперименте планируется изучать воздействие излучения на растения при облучении через k силовых полей. Из заданных u 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 п, /г и описанию u силовых полей определяет, какие *k* силовых полей необходимо выбрать для эксперимента, чтобы площадь участка, покрытого всеми /г силовыми полями, была максимальна, и выводит площадь этого участка.

Формат вхоdлосо *файла*

Первая строка входного файла содержит целые числа п и *k* (I *k \_< n q<* 200 000) общее количество силовых полей и количество силовых полей, которые необходимо выбрать для эксперимента.

Последующие п строк содержат по два целых числа z;, *уж* (l *< х;, уж<* 109) координаты дальнего от начала координат угла прямоугольного участка *i-гo* силового поля.

#### *Формат выходного файла*

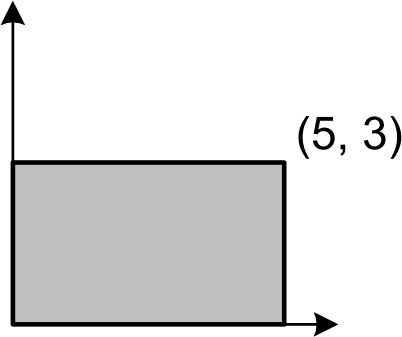
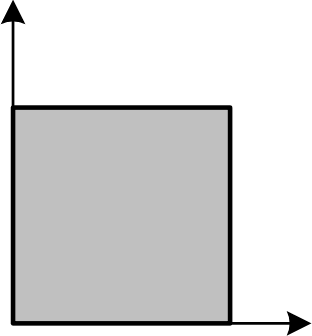
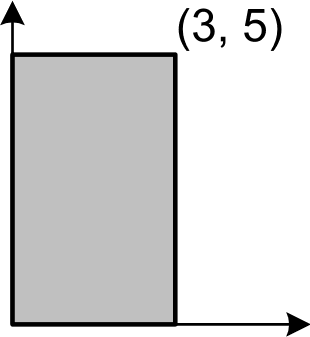
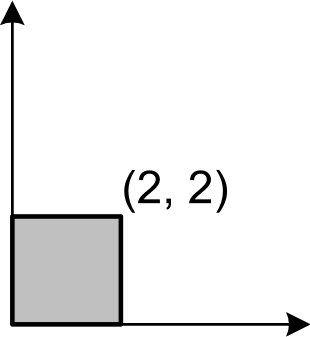
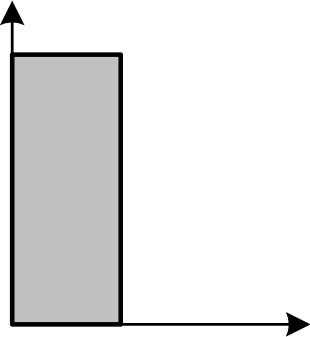
Требуется вывести одно целое число: максимальную площадь искомого участка

*Пример входных и выходных файлов*

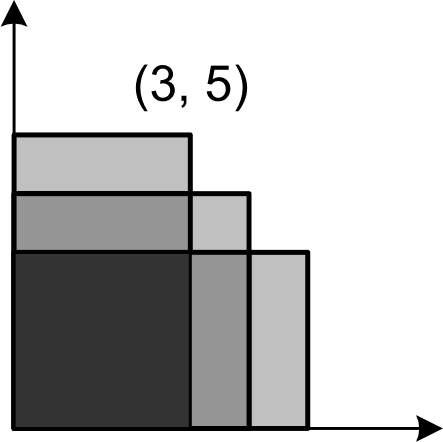
|  |  |
| --- | --- |
| **power.in** | **power.out** |
| 5 3 | 9 |
| 3 5 |  |
| 2 2 |  |
| 2 5 |  |
| 4 4 |  |

Лояслелье х лрьмеру

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

(2, 5)

(4, 4)

Рис 1. Силовые поля в примере описания входных данных.

(4, 4)

(5, 3)

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.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Ограничения** | | Необходимые подзадачи |
| п | *k* |
| 1 | 18 | 1 Е Е 20 | 1 *k п* |  |
| 2 | 25 | 1 u 300 | 1 *k п* | 1 |
| 3 | 20 | 1 Е Е 3 000 | 1 *k п* | 1, 2 |
| 4 | 17 | 2 Е Е 200 000 | *k ——* 2 |  |
| 5 | 20 | 1 < п < 200 000 | 1 < *k < п* | 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: good . 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. Bceгo в компании работают u сотрудников. Каждому сотруднику присвоен уникальный номер от 1 до u, директору присвоен номер 1. У каждого сотрудника, кроме директора, есть ровно один непосредственный начальник. Непосредственный начальник сотрудника *i* имеет номер *ра,* причем *pi < і.*

Сотрудник х является подчиненным уровня 1 сотрудника *у,* если *р —— у.* Для t > 1 сотрудник х является подчиненным уровня k сотрудника у, если сотрудник *р* является подчиненным уровня k — 1 сотрудника у.

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

Перед тем, как выбрать числа Л и Л, директор получил m пожеланий от сотрудников компании, y-e пожелание задается двумя числами п/ и *k;* и означает, что сотрудник п; просит отправить на курсы одного из своих подчиненных уровня *k .* Для экономии средств директор хочет выбрать такие *L* и Л, 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 Л, что если отправить на курсы повышения квалификации всех сотрудников с номерами от *L* до Л 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 u — количество сотрудников компании (2 < п < 200 000). Вторая строка содержит (u — 1) чисел: *р , pз, . .., р* (1 *< ра < i) —* номера непосредственных начальников сотрудников.

Третья строка содержит число m — количество пожеланий от сотрудников.

Последующие ш строк задают пожелания сотрудников и содержат по два целых числа u;, k; (1 < п; < п, 1 < *k < n,* гарантируется, что у сотрудника п; есть хотя бы один подчиненный уровня /г;).

Формат *выходного файла*

Необходимо вывести два искомых числа: *L п R.* Если оптимальных пар *(L, R)*

несколько, требуется вывести ту, в которой значение *L* минимально.

*Пример входных и выходных файлов*

|  |  |
| --- | --- |
| **qual.in** | **qual.out** |
| 7  1 1 2 2 3 3 | 3 6 |
| 1 1 |  |
| 3 1 |  |
| 1 2 |  |

Поясиеиие 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.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Ограничения** | | **Необходимые**  **ОДЗіl іlЧИ** |
|  | **Дополнительные условия** |
| 1 | 19 | 2 < < 50 |  |  |
| 2 | 25 | 2 Е Е 3000 |  |  |
| 3 | 21 | 2 п 200 000 | для всех *i* выполнено *pi* ——*i* — 1 |  |
| 4 | 35 | 2 Е п Е 200 000 |  | 1, 2, 3 |

/7олучелье **ьлформацпь** о *результатах окончательной* **лроверхп**

По запросу сообщаются баллы за каждую подзадачу.