



Муниципальный этап Всероссийской олимпиады школьников 2015/16 учебный год

Предмет	Класс	Дата	Время начала	Время окончания
Информатика и ИКТ	7-8	26.11.2015	10-00	14-00

Задача 1. Олимпиадный комплект

Имя входного файла:	input.txt
Имя выходного файла:	output.txt
Ограничение по памяти:	256 МБ
Ограничение по времени:	1 секунда на тест
Максимальная оценка за задачу:	100 баллов

В некотором государственном университете (далее НГУ) решили провести командную студенческую олимпиаду. В каждой команде три человека.

Для каждой команды необходим компьютер, два стола, три стула, три ручки и три блокнота. Вам известно, сколько в НГУ есть компьютеров, столов, стульев и сколько спонсоры выделили ручек и блокнотов. *Напишите программу*, которая определит, какое максимальное число команд НГУ может пригласить на олимпиаду.

Входные данные

В первой и единственной строке входного файла содержатся пять целых неотрицательных чисел, записанных через пробел — количество компьютеров, столов, стульев, ручек и блокнотов. Все числа не превосходят 10^9 .

Выходные данные

В выходной файл нужно вывести ответ, одно целое число — максимальное число команд, которое НГУ может пригласить на олимпиаду.

Примеры

<i>input.txt</i>	<i>output.txt</i>
3 4 7 9 6	2
3 1 7 9 6	0

Задача 2. Горная цепь

Имя входного файла:	input.txt
Имя выходного файла:	output.txt
Ограничение по памяти:	256 МБ
Ограничение по времени:	1 секунда на тест
Максимальная оценка за задачу:	100 баллов

Хоть это лето и не сильно радовало хорошей погодой, Андрей всё же собрался в очередной поход. На этот раз он решил подготовиться основательно — несмотря на то, что ещё оставалось чуть больше недели, все вещи уже были собраны, остался лишь один небольшой момент — нужно было нарисовать примерную карту предстоящего маршрута. К счастью, Андрей уже хорошо представляет, какие именно вершины и в каком порядке ему предстоит покорить.

Для того, чтобы начертить план, Андрей взял листок в клеточку, имеющий высоту h , и начал заполнять клетки определёнными знаками. Линиями «/» и «\» он решил изобразить очертания самих гор. Символом «.» он решил отобразить небо, то есть всю часть рисунка, которая находится над горами. Символ «#» на его плане обозначает землю под горами. Что



**Муниципальный этап Всероссийской олимпиады школьников
2015/16 учебный год**

Предмет	Класс	Дата	Время начала	Время окончания
Информатика и ИКТ	7-8	26.11.2015	10-00	14-00

касается самих гор, то Андрей пояснил, что их подножье всегда лежит на одной высоте – это и есть самая нижняя строка в его плане. Далее гора поднимается до некоторой высоты и идет вниз, к подножью следующей горы, после чего начинается следующая гора, если она есть. По длине Андрей решил обрезать листок таким образом, чтобы последний столбец клеточек обозначал последнюю вершину, которую он намеревается покорить. Так, например, если высота листка h равна 5, а высоты первой, второй и третьей вершин, которые он хотел бы посетить, равны 3, 1 и 4, соответственно, то в конце концов он получит следующий рисунок:

```

.....
...../\...
../\...../##\..
./##\...../####\
/####\//#####\

```

Обычно Андрей чертил такие планы от руки, но так как в этот раз он подошёл к подготовке основательно, то и план решил распечатать на принтере, с чем и возникли определённые трудности.

Андрей, как один из ваших лучших друзей, понимает, что вы готовы разделить с ним все сложности, возникающие при выполнении этой задачи, именно поэтому он и обратился к вам за помощью в *написании программы*. Для наглядности, он привёл ниже пример маршрута, который был в прошлом году.

Входные данные

В первой строке входного файла записано два целых числа: h и n – высота клетчатого листка и количество вершин, которое намеревается покорить Андрей ($2 \leq n, h \leq 100$).

Во второй строке записана последовательность целых чисел a_1, a_2, \dots, a_n – высота первой, второй, ..., n -й вершины, соответственно, в том порядке, в котором Андрей хотел бы посетить эти вершины ($1 \leq a_i \leq h$).

Выходные данные

В выходной файл требуется вывести маршрут Андрея в описанном выше формате.

Пример

<i>input.txt</i>	<i>output.txt</i>
<pre> 5 3 3 1 4 </pre>	<pre>/\... ../\...../##\.. ./##\...../####\ /####\//#####\ </pre>



**Муниципальный этап Всероссийской олимпиады школьников
2015/16 учебный год**

Предмет	Класс	Дата	Время начала	Время окончания
Информатика и ИКТ	7-8	26.11.2015	10-00	14-00

Задача 3. Лесополоса

Имя входного файла:	input.txt
Имя выходного файла:	output.txt
Ограничение по памяти:	256 МБ
Ограничение по времени:	1 секунда на тест
Максимальная оценка за задачу:	100 баллов

И вот долгожданный отпуск. Шпион Виталий упаковал чемоданы и отправился в путешествие на поезде. Однако поездка оказалась не столь увлекательной, как предполагал Виталий, вместо бескрайних просторов необъятной Родины из окна вагона видна была одна лишь лесополоса.

«Скукота, да и только», подумал Виталий и уже начал готовиться ко сну, как вдруг неожиданность. Прямо по курсу ремонт дороги. Поезд снизил скорость, за окном перестали проноситься верхушки деревьев, огромное количество дорожных ремонтников по обе стороны от путей... Самое время потренировать свои профессиональные навыки, подумал Виталий.

Начать, пожалуй, можно с того, что правильно посчитать количество рабочих с одной и с другой стороны от путей. Но ведь как-то нужно будет и проверить себя, подумал Виталий.

В его арсенале была хорошая камера и алгоритм, распознающий образы людей и представляющий их в виде точек на плоскости. Но, к сожалению, алгоритм не умеет понимать, с левой или с правой стороны от путей находятся люди. Мало того, помимо рабочих за окном находятся и обычные пешеходы. К счастью, известно, что они находятся за барьерами "ремонт дороги", которые алгоритм распознаёт как две точки на плоскости, через которые этот барьер проходит. Помимо этого алгоритм умеет распознавать железнодорожные пути, их он тоже распознаёт две точки на плоскости, через которые проходят пути. Барьеры всегда параллельны железнодорожным путям.

Таким образом, вам остаётся лишь *написать программу*, которая посчитает, сколько людей находится по правую и левую стороны от железнодорожных путей, стоящих не дальше прямых, ограничивающих ремонт дороги. Левая и правая сторона от путей определяется по ходу движения поезда. Гарантируется, что ни на рельсах, ни на барьерах людей нет.

Входные данные

В первой строке входного файла записаны четыре целых числа: x_1, y_1, x_2, y_2 – координаты точек на железнодорожных путях ($-10^4 \leq x_1, y_1, x_2, y_2 \leq 10^4$). Заданы они таким образом, что направление движения поезда совпадает с направлением из точки (x_1, y_1) в точку (x_2, y_2) .

Во второй строке записаны четыре целых числа: xa_1, ya_1, xa_2, ya_2 – координаты точек на барьере слева от путей по ходу движения ($-10^4 \leq xa_1, ya_1, xa_2, ya_2 \leq 10^4$). Заданы они таким образом, что направление из точки (xa_1, ya_1) в точку (xa_2, ya_2) совпадает с направлением движения поезда.

В третьей строке записаны четыре целых числа: xb_1, yb_1, xb_2, yb_2 – координаты точек на барьере справа от путей по ходу движения ($-10^4 \leq xb_1, yb_1, xb_2, yb_2 \leq 10^4$). Заданы они тоже таким образом, что направление из точки (xb_1, yb_1) в точку (xb_2, yb_2) совпадает с направлением движения поезда. Гарантируется, что железнодорожные пути не совпадают с барьерами.

В четвёртой строке содержится единственное число N – количество людей, распознанных алгоритмом ($1 \leq N \leq 2 \cdot 10^5$).



Муниципальный этап Всероссийской олимпиады школьников
2015/16 учебный год

Предмет	Класс	Дата	Время начала	Время окончания
Информатика и ИКТ	7-8	26.11.2015	10-00	14-00

В следующих N строках записаны пары целых чисел x_i, y_i – координаты i -го человека на плоскости. Все координаты по модулю не превосходят 10^4 , в одной точке может находиться только один человек.

Выходные данные

В выходной файл требуется вывести в одной строке через пробел два целых числа – количество рабочих, находящихся слева от железнодорожных путей по ходу движения поезда и справа от них, соответственно.

Примеры

<i>input.txt</i>	<i>output.txt</i>
0 0 1 1 -2 0 0 2 0 -2 2 0 7 -4 -1 0 3 2 3 0 1 -2 -3 0 -1 2 -3	2 2
1 0 1 1 -5 0 -5 1 3 0 3 1 8 -2 -2 -1 5 -3 6 -4 4 2 -2 2 1 2 4 4 6	4 3