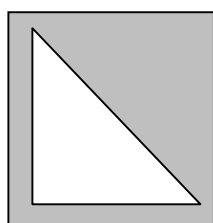


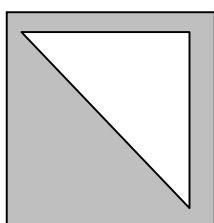
Задача 1. POBEDA-2014

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

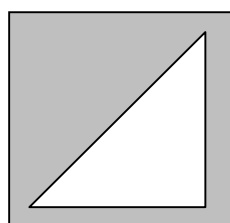
Как известно, современные видеокарты умеют формировать изображения с использованием только треугольников. Видеокарта POBEDA-2014 не отстает от современных тенденций. Известно, что она умеет отображать только прямоугольные равнобедренные треугольники четырех типов ориентации, представленные на рисунках ниже. Изменять ориентацию этих треугольников видеокарта не может.



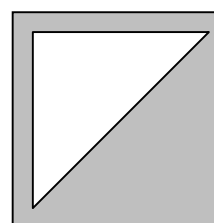
1 тип



2 тип



3 тип



4 тип

Длина катета каждого из представленных выше треугольников равна одному сантиметру. За один такт видеокарта не может отобразить более чем a_i треугольников i -того типа.

Необходимо определить максимально возможную длину стороны квадрата, который может быть изображен видеокарткой на экране монитора за один такт. При этом квадрат должен быть расположен так, чтобы его стороны были параллельны краям монитора.

Требуется написать программу, которая решает поставленную задачу.

Формат входного файла

Первая строка входного файла содержит разделенные пробелами четыре целых числа: a_1, a_2, a_3, a_4 ($0 \leq a_1, a_2, a_3, a_4 \leq 10^{18}$). Входные данные могут превышать максимальные значения для 32 битного типа данных.

Формат выходного файла

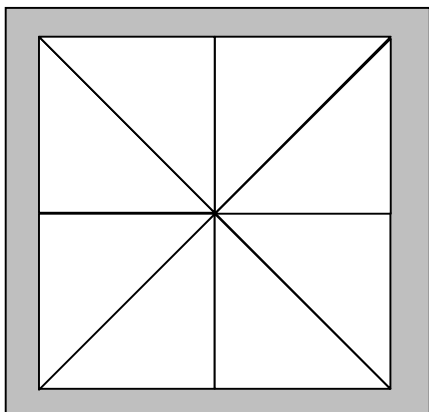
Выходной файл должен содержать одно число – максимально возможную длину стороны квадрата.

Примеры входных и выходных файлов

<code>pobeda.in</code>	<code>pobeda.out</code>
2 2 2 2	2
10 10 0 0	3

Пояснения к примерам

Далее приведен рисунок для первого примера.



Система оценивания

Частичные правильные решения для тестов, в которых $a_1, a_2, a_3, a_4 \leq 100\,000$, будут оцениваться из 50 баллов.

Задача 2. Список школ

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

При регистрации на портале интернет-олимпиады все участники заполняют регистрационную форму, где они указывают название школы, в которой они учатся. Разные участники могут по-разному писать название школы, например, «Физико-математическая школа №18», «ФМШ №18».

Организаторам олимпиады предоставлена информация о названиях школ, которые написали регистрируемые участники олимпиады. Точно известно, что цифры в названии школы встречаются только в номере школы, а число в записи названия школы встречается ровно один раз и оно однозначно определяет номер школы. Номер школы является положительным целым числом и не может начинаться с нуля.

Требуется написать программу для сайта интернет-олимпиады, которая поможет организаторам олимпиады получить следующую информацию: количество школ и номера школ, из которых зарегистрировалось не более пяти участников.

Формат входного файла

Первая строка входного файла содержит одно целое число n ($1 \leq n \leq 1000$) – количество названий школ, указанных всеми участниками при регистрации.

Последующие n строк содержат названия школ, указанные всеми участниками. Название школы содержит только заглавные и строчные буквы латинского алфавита, цифры и пробелы, длина названия не превышает 100 символов.

Формат выходного файла

Первая строка выходного файла должна содержать одно число m – количество школ, от которых на олимпиаду зарегистрировалось от одного до пяти участников. Последующие m строк должны содержать только номера таких школ, при этом номера должны располагаться по одному в строке в произвольном порядке.

Примеры входных и выходных файлов

<code>schools.in</code>	<code>schools.out</code>
9	2
Physics and Mathematics School 18	42
9ya shkola imeni Pushkina	18
Lyceum 9	
PaMS 18	
Gymnasium 42	
School 9	
Shkola nomer 9	
High school 9	
School N 9	

Пояснения к примерам

В приведенном примере для участия в интернет-олимпиаде зарегистрировались: два ученика из школы с номером 18, один ученик из школы с номером 42 и шесть учеников из школы с номером 9. Таким образом, от 1 до 5 участников зарегистрировано от школ с номерами 18 и 42.

Система оценивания

Частичные правильные решения для тестов, в которых все номера школ являются однозначными числами, будут оцениваться из 30 баллов.

Частичные правильные решения для тестов, в которых номера школ – это числа строго меньше 1000, будут оцениваться из 50 баллов.

Частичные правильные решения для тестов, в которых номера школ – это числа строго меньше 10^9 , будут оцениваться из 80 баллов.

Несмотря на выделение отдельных групп тестов, на окончательную проверку будут приниматься только решения, правильно работающие для теста из условия задачи.

Задача 3. Межрегиональная олимпиада

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

На межрегиональной олимпиаде по программированию роботов соревнования проводятся в один тур и в необычном формате. Задачи участникам раздаются последовательно, а не все в самом начале тура, и каждая i -я задача ($1 \leq i \leq n$) становится доступной участникам в свой момент времени s_i . При поступлении очередной задачи каждый участник должен сразу определить, будет он ее решать или нет. В случае, если он выбирает для решения эту задачу, то у него есть t_i минут на то, чтобы сдать ее решение на проверку, причем в течение этого времени он не может переключиться на решение другой задачи. Если же участник отказывается от решения этой задачи, то в будущем он не может к ней вернуться. В тот момент, когда закончилось время, отведенное на задачу, которую решает участник, он может начать решать другую задачу, ставшую доступной в этот же момент, если такая задача есть, или ждать появления другой задачи. При этом за правильное решение i -й задачи участник получает c_i баллов.

Артур, представляющий на межрегиональной олимпиаде один из региональных центров искусственного интеллекта, понимает, что важную роль на такой олимпиаде играет не только умение решать задачи, но и правильный стратегический расчет того, какие задачи надо решать, а какие пропустить. Ему, как и всем участникам, до начала тура известно, в какой момент времени каждая задача станет доступной, сколько времени будет отведено на ее решение и сколько баллов можно получить за ее решение. Артур является талантливым школьником и поэтому сможет успешно решить за отведенное время и сдать на проверку любую задачу, которую он выберет для решения на олимпиаде.

Требуется написать программу, которая определяет, какое максимальное количество баллов Артур сможет получить при оптимальном выборе задач, которые он будет решать, а также количество и перечень таких задач.

Формат входного файла

Первая строка входного файла содержит одно целое число n ($1 \leq n \leq 100\,000$) - количество задач на олимпиаде.

Последующие n строк содержат описания задач, по три числа на каждой строке: s_i - момент появления i -й задачи в минутах, t_i - время, отведенное на ее решение в минутах, и c_i - сколько баллов получит участник за решение этой задачи ($1 \leq s_i, t_i, c_i \leq 10^9$).

Формат выходного файла

Первая строка выходного файла должна содержать одно число - максимальное количество баллов, которое сможет получить Артур на олимпиаде.

Вторая строка должна содержать одно целое число m - количество задач, которые надо решить при оптимальном выборе.

Третья строка должна содержать m разделенных пробелом целых чисел - номера этих задач в порядке их решения. Задачи пронумерованы, начиная с единицы, в порядке их описания во входном файле.

Если оптимальных ответов несколько, необходимо вывести любой из них.

Примеры входных и выходных файлов

<code>olympiad.in</code>	<code>olympiad.out</code>
2	3
1 1 1	2
2 2 2	1 2
3	3
1 2 1	1
3 2 1	3
2 4 3	

Пояснения к примерам

В первом примере Артур успеваеt решить все задачи и получить три балла.

Во втором примере Артуру выгоднее решать последнюю задачу и получить за нее три балла, чем решать только первые две и получить два балла.

Система оценивания

Частичные правильные решения для тестов, в которых все c_i одинаковы и $n \leq 1000$, оцениваются из 30 баллов.

Частичные правильные решения для тестов, в которых все c_i одинаковы, оцениваются из 50 баллов.

Частичные правильные решения для тестов, в которых $n \leq 1000$, оцениваются из 50 баллов.

Задача 4. Дом Мэра

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

При планировании нового района города М было решено, что дороги в новом районе будут образовывать прямоугольную сетку, то есть все улицы будут одного из двух типов – направленные с юга на север или направленные с востока на запад. При этом параллельные улицы будут проходить через каждый километр, и каждый квартал будет иметь размер ровно один километр на один километр. Таким образом, все дороги образуют равномерную клетчатую сетку. По каждой дороге разрешен проезд в любом из двух направлений.

Через некоторое время после постройки дорог оказалось, что такая планировка не всегда удобна, поскольку для постройки больших заводов или других сооружений и организации парков недостаточно одного квартала. Мэрия города М решила отдать каждому большому проекту по прямоугольному блоку из нескольких соседних кварталов. К сожалению, после реализации проекта все дороги внутри такого блока будут закрыты для проезда, но по границе блоков проезд все еще будет возможен. Касание двух блоков не закрывает проезд между ними.

Когда Мэру города М принесли на согласование план распределения территорий для больших проектов, ему стало интересно, насколько сложным будет маршрут от мэрии до его будущего дома. Мэрия находится в центре нового района, на пересечении нулевой улицы, направленной с юга на север, и нулевой улицы, направленной с востока на запад. С итоговым расположением дома Мэр еще не определился и на выбор у него есть k вариантов. Каждый из вариантов находится на пересечении x_i -ой улицы, направленной с юга на север (положительный x означает, что улица находится восточнее мэрии, отрицательный – западнее) и y_i -ой улицы, направленной с востока на запад (положительный y означает, что улица находится севернее мэрии, отрицательный – южнее).

Мэр считает, что маршрут до дома является сложным, если ему на этом маршруте придется совершить более двух поворотов направо или налево. Машина Мэра не может совершать более одного поворота на перекрестке, например, чтобы развернуться. Длина маршрута не имеет значения, и к дому можно подъезжать с любой стороны. Машина Мэра всегда стоит у мэрии в северном направлении, может повернуть сразу направо или налево, но не может развернуться.

Требуется написать программу, которая по информации о закрытых для проезда блоках кварталов и возможным расположениям дома Мэра, для каждого возможного расположения дома Мэра найдет несложный маршрут от мэрии до дома, определит кратчайший из них, или сообщит, что такого маршрута не существует. Количество поворотов минимизировать не требуется.

Формат входного файла

Первая строка входного файла содержит два целых числа n и k ($0 \leq n \leq 100\,000$, $1 \leq k \leq 10$) – количество блоков кварталов, которые по плану будут отданы большим проектам и количество вариантов расположения дома Мэра, соответственно.

Последующие n строк содержат по описанию блоков кварталов – четыре целых числа u_1, v_1, u_2, v_2 ($-10^9 \leq u_1 < u_2 \leq 10^9$, $-10^9 \leq v_1 < v_2 \leq 10^9$) — номера улиц, на пересечении которых расположены противоположные углы блока кварталов, отданных под застройку и закрытых для проезда.

Последующие последние k строк содержат по два целых числа x_i и y_i ($|x_i| \leq 10^9$, $|y_i| \leq 10^9$, $x_i \neq 0$ или $y_i \neq 0$) — возможные расположения дома Мэра.

Мэрия и никакое из возможных расположений дома Мэра не находятся внутри блоков кварталов, отданных под застройку, но блоки кварталов, отданные под застройку, могут пересекаться.

Формат выходного файла

В выходной файл для каждого из возможных расположений дома Мэра в порядке появления во входном файле необходимо вывести сообщение, существуют ли несложный маршрут от мэрии до дома Мэра и, если существует, то где надо сделать повороты.

Если не существует несложный маршрут, то сообщение должно содержать слово NO на одной строке. Иначе, в первой строке должно содержаться слово YES, во второй строке — одно число t ($0 \leq t \leq 2$) — количество поворотов, и в последующих t строках — описания поворотов в порядке их совершения: в каждой строке по три числа x , y и d — номера улиц, на которых расположен перекресток, где необходимо повернуть, и направление поворота, при этом $d = -1$ означает поворот налево и $d = 1$ — поворот направо.

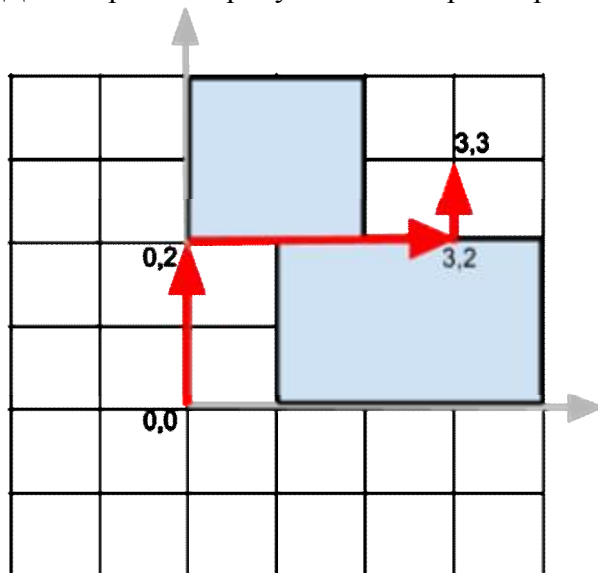
Координаты перекрестков, где необходимо совершить повороты, не должны превышать 10^9 . Если кратчайших несложных путей несколько, необходимо вывести любой из них.

Примеры входных и выходных файлов

majorhouse.in	majorhouse.out
1 2 -2 1 9 2 2 0 3 3	YES 1 0 0 1 NO
2 1 0 2 2 4 1 0 4 2 3 3	YES 2 0 2 1 3 2 -1
0 2 0 -1 0 1	NO YES 0

Пояснения к примерам

Далее приведен рисунок для второго примера.



Система оценивания

Частичные правильные решения для тестов, в которых все координаты (x , y , u и v) по модулю не превышают 100, и $n \leq 50$, будут оцениваться из 30 баллов.

Частичные правильные решения для тестов, в которых $n \leq 50$, будут оцениваться из 60 баллов.