



Задача А. ЛЛШ

Имя входного файла:	<code>stdin</code>
Имя выходного файла:	<code>stdout</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Как вы наверное знаете, существует страна под названием Лайнландия (Lineland). Эта страна во многом похожа на другие страны, но есть одно отличие, которое заключается в том, что эта страна одномерная и представляет собой прямую.

В Лайнландии также, как и у нас проходят различные летние школы (Лайнландцы ведь тоже люди). Например, каждый год там проходит широко известная в линейных кругах ЛЛШ (Летняя Лайнландская Школа). Именно в этой школе и происходит действие нашей задачи.

В этом лагере работает вожатым герой нашей истории, Вениамин. К великому для него и его соседа по комнате, которого зовут Э, сожалению, при подготовке к заезду они заселились в комнату без замка. Поэтому им приходится прятать всю еду, которую они отбирают у детей, в тайнике под деревом. (точней Вениамин прячет, а Э. контролирует :)

Еще, как и во всяком лагере, в ЛЛШ бывают пожарные проверки. На них обычно очень весело, сначала все бегут к автобусной остановке, крича «Да здравствует пожарная тревога!», а потом тушат флагшток (ну это как повезет).

Еще иногда бывает проверка СЭС. Она проходит уже не так весело. Работники СЭС отличаются от пожарников тем, что на работе их не кормят, поэтому во время таких проверок приезжает n СЭСовцев, главная задача которых состоит в том, чтобы исследовать лагерь для того, чтобы найти и съесть всю одномерную еду, которую они могут съесть. Так как СЭСовцы очень давно не ели (с прошлого сезона), то можно считать, что вся еда, которая была спрятана на исследованном ими отрезке будет обнаружена и уничтожена. Искать и употреблять в пищу двухмерную, трехмерную и четырехмерную еду они не умеют, хотя очень хотели бы. Хм, да эти неудачники еще не слышали о проективном пространстве...

Так как лагерь очень большой (директор на открытии говорил, что он простирается аж на отрезке $(0; len)$ в одной из систем координат Лайнландии), а СЭСовцы не всегда оптимально выбирают отрезки для исследований, то у Вени есть надежда, что его тайник не найдут. Однако Э. хочет в этом убедиться наверняка. Поэтому он отправил учеников следить за СЭСовцами, мотивировав их задачей про Кузнечика, Который Смог. По результатам наблюдений Э. и Вениамин узнали, какой отрезок исследовал каждый СЭСовец. Э. разумно заметил, что Лайнландия расположена в хаусдорфовом пространстве, а значит любые его две точки обладают непересекающимися окрестностями. Теперь Вени, используя этот мудрый факт, предстоит выяснить суммарную длину участков лагеря, где еда была уничтожена СЭСовцами. Но, так как Вениамин лишь блистательный геометр, а программист из него не очень, то он просит вас помочь.

Формат входного файла

В первой строке заданы два целых числа n и len ($2 \leq n \leq 10^5, 1 \leq len \leq 10^{15}$) — количество проверяющих и длина лагеря.

Далее, в n строках заданы отрезки, исследованные проверяющими. i -ый отрезок задан целыми координатами его начала и конца l_i и r_i ($0 \leq l_i < r_i \leq len$).

Формат выходного файла

Вам нужно вывести одно число – суммарную длину участков лагеря, проверенных СЭСовцами.



Примеры

stdin	stdout
4 7	5
1 2	
1 4	
3 5	
6 7	

Пояснение

Если вы не знаете, что такое хаусдорфово пространство, то можете считать, что это обычное пространство.

Тесты к этой задаче состоят из шести групп:

- Тест 1. Тест из условия. Оценивается в 0 баллов.
- Тесты 2 – 11. В тестах этой группы $n = 2$. Оцениваются в 20 баллов.
- Тесты 12 – 16. В тестах этой группы $n = 3$. Оцениваются в 10 баллов.
- Тесты 17 – 36. В тестах этой группы $n \leq 10^4, len \leq 10^4$. Оцениваются в 40 баллов.
- Тесты 37 – 41. В тестах этой группы $n \leq 10^4, len \leq 10^6$. Оцениваются в 10 баллов.
- Тесты 41 – 51. В тестах этой группы дополнительные ограничения отсутствуют. Оцениваются в 20 баллов.



Задача В. Кузнечик, Который Смог

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Как мы вам уже говорили, для того чтобы мотивировать детей, Э. рассказал им историю про Кузнечика, Который Смог (ККС). Если вы ее не знаете, то она о кузнечике, который не мог достичь своей цели, однако он изо всех сил стремился к ней, все ближе приближаясь к ней (или, если проще, $\forall \varepsilon > 0 \exists t : d(x(t), x_0) < \varepsilon$). А потом, чтобы взять с него пример, они все вместе доказали, что он смог.

Однако мы вам расскажем историю про другого маленького кузнечика, который очень хочет стать таким же, как Тот Кузнечик, когда вырастет. Кузнечик живет в стране Циркулусландия, так же известную как Z_n -ландия. Эта страна представляет собой окружность, разбитую на n дуг, между которыми ее жители могут перемещаться. Дуги нумеруются с 1 против часовой стрелки. Наш герой живет в дуге с номером e . Так как в Циркулусландии нет нефти, и демократию туда принести некому, то ее жителям приходится мириться с суровым цикадократическим режимом. Например, для перемещения по стране жители должны использовать талоны на путешествия. Один талон дает право переместиться на ровно k дуг по часовой или против часовой стрелки. При этом в одной дуге без предварительного согласования с властями не может собираться более трех кузнечиков.

Однажды герой нашей истории, Кузнечик, Который Хочет Когда-нибудь Смочь, ушел гулять, но (по некоторым версиям) подскользнулся, упал и потерял сознание... Когда он очнулся, он обнаружил, что находится в дуге с номером b , у него осталось m талонов на путешествие и ему нужно срочно попасть домой, поскольку его часы показывали, что пришло время пить чай. Ваша задача состоит в том, чтобы сказать, сможет ли он попасть домой, не нарушая при этом закон и не грабя прохожих.

Формат входного файла

В первой строке заданы два числа n и m ($1 \leq n \leq 1000, 1 \leq m \leq 10^5$) — количество дуг в Циркулусландии и количество талонов у Кузнечика, Который Хочет Когда-нибудь Смочь.

Во второй строке два числа b и e — дуга, в которой сейчас находится Кузнечик и дуга, в которую ему нужно вернуться.

Далее, в m строках идут целые числа k_i — количество дуг, на которое дает право переместиться i -ый билет ($1 \leq k_i \leq n - 1$). Один билет нельзя использовать больше одного раза.

Формат выходного файла

Если Кузнечик сможет, выведите в первой строке "YES" (без кавычек). Во второй строке выведите m чисел, где i -ое число равно 0, если i -ый билет не был использован, 1, если он был использован для перемещения против часовой стрелки и -1 , если для перемещения по часовой. Если существует несколько способов, выведите любой.

Если Кузнечик не сможет и ему придется нарушать закон и грабить прохожих, выведите "NO".

Примеры

stdin	stdout
10 4	YES
3 5	-1 1 0 -1
9	
3	
7	
2	



Пояснение

В данной задаче e — целое число и оно вовсе не равно основанию натурального логарифма (2.71828182846)

Тесты к этой задаче состоят из пяти групп:

- Тест 1. Тест из условия. Оценивается в 0 баллов.
- Тесты 2 – 6. В тестах этой группы $m = 2$. Оцениваются в 10 баллов.
- Тесты 7 – 31. В тестах этой группы $m \leq 15$. Оцениваются в 50 баллов.
- Тесты 31 – 41. В тестах этой группы $m \leq 100, n \leq 1000$. Оцениваются в 20 баллов.
- Тесты 41 – 51. В тестах этой группы дополнительные ограничения отсутствуют. Оцениваются в 20 баллов.



Задача С. Метро

Имя входного файла:	<code>stdin</code>
Имя выходного файла:	<code>stdout</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Многие знают, что метро это самый удобный и быстрый общественный транспорт (конечно там, где оно вообще есть). Но к нашему главному герою изобретателю Акакию пришла замечательная мысль, а именно проект под названием «Метро будущего». Он уже давно заметил, что время проведенное в метро прямо пропорционально длине ветки, так как с ростом длины веток растет и время, затрачиваемое поездом на то, чтобы проехать по этой ветке. Поэтому он в своем проекте решил свести длину путей к минимуму. И в результате долгих размышлений он пришел к выводу, что метро должно состоять из одной станции, и тогда суммарная длина веток будет равна нулю. Акакий хотел бы уменьшить ее еще сильнее, но у него пока что не получилось. Но, так как метро должно быть доступно из любой точки города, он решил сделать еще несколько спусков под землю и соединить их между собой и со станцией подземными переходами (у нас даже есть информация, что в одном из городов России этот проект уже воплощают в жизнь).

Акакий создал идеальный план метрополитена так, чтобы из каждого спуска можно было добраться до каждого, но строители были мигрантами из какой-то далекой страны, где слово «метро» обозначает подземелье, созданное для того, чтобы запускать туда людей и пытаться их. Поэтому они постарались сделать его как можно лучше, и в эксплуатацию пошло метро без части переходов. Акакий был в ярости, когда узнал про это, но ничего не поделаешь нужно было учитывать особенности перевода...

Теперь он хочет знать лишь то, можно ли добраться от спуска около его дома до спуска рядом с его работой по подземным переходам. Но схема настолько сложна, что вручную ее не разобрать, поэтому он решил обратиться к вам за помощью. Ах да, Акакий уже гражданин преклонных лет, из-за чего он уже не помнит, где живет и работает, поэтому вам передадут t запросов, для каждого из которых вам нужно будет дать ответ.

Формат входного файла

В первой строке входного файла заданы два числа n и m ($1 \leq n \leq 1000, 1 \leq m \leq \frac{n(n-1)}{2}$) — количество спусков под землю и количество переходов.

Далее, в m строках идут числа f_i и t_i ($1 \leq f_i, t_i \leq n$), номера спусков, который этот переход соединяет. Сама станция совмещена со спуском 1, который остался после ее строительства.

Далее идет число t — количество запросов ($t \leq 10^5$). Далее в каждый запрос задается двумя числами — начальным и конечным спуском.

Формат выходного файла

Для i -го запроса в i -ой строке выведите "YES" (без кавычек), если Акакий сможет добраться от дома до работы, иначе "NO".



Примеры

stdin	stdout
6 4	YES
1 2	NO
1 3	YES
2 3	NO
5 4	YES
5	
3 2	
6 1	
4 5	
1 5	
1 2	

Пояснение

Тесты к этой задаче состоят из пяти групп:

- Тест 1. Тест из условия. Оценивается в 0 баллов.
- Тесты 2 – 16. В тестах этой группы к каждому спуску проведено не более двух переходов. Переходы не могут образовывать циклы. $t \leq 10$. Оцениваются в 30 баллов.
- Тесты 17 – 31. В тестах этой группы к каждому спуску проведено не более двух переходов. $t \leq 10$. Оцениваются в 30 баллов.
- Тесты 31 – 41. В тестах этой группы $n \leq 100, t \leq 10$. Оцениваются в 20 баллов.
- Тесты 41 – 51. В тестах этой группы дополнительные ограничения отсутствуют. Оцениваются в 20 баллов.



Задача D. Тролли и кинематика

Имя входного файла:	<code>stdin</code>
Имя выходного файла:	<code>stdout</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Возможно вы помните задачу про троллей и Кешу со школьного этапа. В этот раз тролли хотят принести в жертву печеньку. Для этого нужно зарядить ее комплексной энергией, а затем поместить ее в алтарь в виде огромной древней печенки (она уже испортилась, но тролли по-прежнему считают ее священной, потому что у них изысканный вкус, в отличие от некоторых). Для этого у троллей есть две стены, образующие угол α (который как вы уже догадались тоже священный, я бы даже сказал, у троллей все что связано с печенками и стенами священо) и собственно алтарь (стены бесконечные, ну и скажите какие они... да священные, правильно, молодцы!). При этом одна стена заряжает печенку мнимой составляющей энергии, а другая — действительной. Тролли накладывают на печенку заклинание левитации и кидают ее. Так как законы физики действуют и в этой вселенной, то при столкновении со стенками печенка упруго от них отгаликивается. Это значит, что угол ее падения на стенку равен углу отражения, а величина ее скорости не меняется (см. картинку в примечаниях). Так как трение между печенкой и комплексной энергией практически отсутствует, то вращением печенки вокруг своей оси можно пренебречь. Заклинание левитации действует так: печенка горизонтально летит, а потом через t секунд внезапно падает (ну при учете того, что t все знают, внезапным падение не назовешь, но ритуал священен, поэтому тролли обязаны удивляться действиям печенки).

Наученный горьким опытом Иннокентий сидит на дереве и наблюдает за троллями. Поскольку с оторванной рукой очень сложно залезать на дерево (нам самим непонятно, как ему это удалось, но это уже совсем другая задача), то бинокль остался внизу. Кеша уже давно наблюдает за троллями, поэтому знает значение t . Так же он знает начальные координаты печенки и координаты ее вектора скорости. Кеше очень важно знать, куда упадет печенка, потому что по легенде ритуальное поедание этой печенки вызывает отрастание утерянных конечностей (правда есть вероятность, что при этом еще лишние отрастут, но это уже не так страшно). Поэтому ему нужно просчитать траекторию полета печенки и узнать координаты места ее падения. Тогда наш герой сможет вернуться домой с целыми конечностями, а иначе его накажет мама, которая сказала, что независимо от обстоятельств, Кеша не должен терять своих рук. Но, так как Кешин калькулятор лежит рядом с его биноклем, и он не может сам просчитать траекторию полета печенки, то он попросил это сделать Вас (кто Вы вообще такие, что все Вас просят что-то посчитать?).

Формат входного файла

В первых трех строках записано три пары действительных чисел $x_A, y_A, x_O, y_O, x_B, y_B$ — координаты точки на первой стене, вершины угла и точки на второй стене соответственно (все числа лежат в интервале $(-100; 100)$).

Во второй строке задано действительное число t ($0 < t \leq 100$).

В третьей строке заданы начальные координаты печенки x и y , в четвертой — координаты вектора ее скорости v_x и v_y .

Формат выходного файла

В единственной строке выведите два числа — координаты места падения печенки. Ответ засчитывается при абсолютной или относительной погрешности не превышающей 10^{-6}

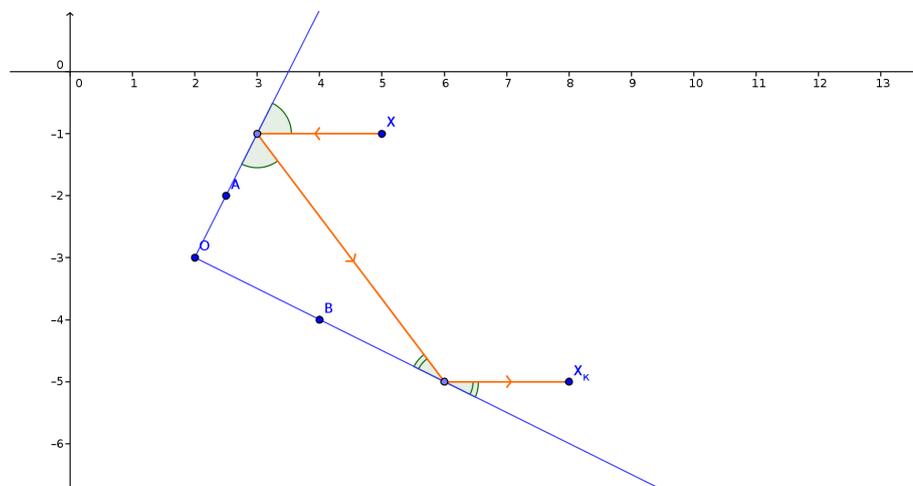


Примеры

stdin	stdout
2.5 -2	8.00 -5.00
2 -3	
4 -4	
4.5	
5 -1	
-2 0	

Пояснение

Расположение стен и траектория полета в первом примере:



Тесты к этой задаче состоят из пяти групп:

- Тест 1. Тест из условия. Оценивается в 0 баллов.
- Тесты 2 – 31. В тестах этой группы стены совпадают с осями координат Ox и Oy . Оцениваются в 60 баллов.
- Тесты 31 – 51. В тестах этой группы дополнительные ограничения отсутствуют. Оцениваются в 40 баллов.



Задача Е. Робот

Имя входного файла:	<code>stdin</code>
Имя выходного файла:	<code>stdout</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Аркадий (вы наверное про него слышали) мечтает захватить мир (впрочем уже не в первый раз). Но, так как у него в одиночку это скорее всего не получится, он решил сделать и запрограммировать робота, который ему в этом поможет. Пока что он еще не закончил разработку своего робота (легко догадаться почему, ведь мир еще не принадлежит ему :)), он только научил его выполнять команды `push_up()`, `push_down()`, `push_left()` и `push_right()`, а именно он просто научил робота выполнять эти команды в определенном порядке. Каждая команда дает роботу указание идти до тех пор, пока он может (пока не уперся в стенку), например, при исполнении команды `push_up()` робот будет идти вверх пока не встретит стенку (команды `push_down()`, `push_left()` и `push_right()` приказывают роботу идти соответственно вниз, влево и вправо). Аркадий собирается штурмовать подземелье и для этого ему нужен робот. Так как он тщательно готовится к штурму, то у него уже есть план этого подземелья, который он купил в переходе за 7 рублей 40 копеек (и нет, это не план метро из 3 задачи). На его плане подземелье представляет собой прямоугольник, разбитый на квадратные клетки (n клеток в высоту и m клеток в ширину). По периметру, а так же между некоторыми клетками расположены стенки, которые нельзя ломать (потому что потом придут тролли и надают ему по шее за такое обращение с священными стенами. По словам Кеши, это не очень приятно). Так как ему возможно придется оттуда сваливать (вернее ему точно придется оттуда сваливать, потому что, узнав о взломе, бравые ребята из ИВФ захотят повязать Аркадия и внести в его жизнь немного демократии), а сваливать он будет до вертолетной площадки, находящаяся на клетке $(1; 1)$, где его ждет вертолет (который конечно же существует лишь в его большой голове). В общем он хочет, как все крутые парни смыться от федералов (ой... ну в смысле от ИВФников), и да, потом еще подземелье должно быть уничтожено большим взрывом, на который Аркадий не будет смотреть (как и все крутые парни, кроме Джанго). Но робот слишком тяжел, чтобы наш герой тащил его на себе, и слишком дорог его сердцу, чтобы бросить его, поэтому Аркадий решил заранее сделать для робота программу, чтобы он тоже мог свалить самостоятельно. Однако, так как он точно не знает, из какой точки подземелья ему придется сваливать, он решил сделать универсальную программу (создать универсальную программу в его понимании – накидать ровно k рандомных команд и надеяться, что робот выберется). Теперь ему интересно количество точек подземелья, из которых эта программа сможет вывести робота на вертолетную площадку. Уважаемые ребята, помогите Дане найти жулика Аркадию найти ответ на этот вопрос.

Формат входного файла

В первой строке заданы два целых числа n и m ($1 \leq n, m \leq 500$) — ширина и высота карты подземелья.

Далее, во второй строке задано количество вертикальных стенок w_v (не считая стенки по краям подземелья). В следующих w_v строках идут координаты клеток, у которых слева есть стенка.

Затем идет количество горизонтальных стенок w_h , и в следующих w_h строках идут координаты клеток, под которыми есть стенка.

Далее идет количество команд в программе робота k ($1 \leq k \leq 1000$). В следующих k строках идут команды, по одной команде в строке. Команды имеют вид "`push_up()`"; (см. пример)

Формат выходного файла

Вам нужно вывести одно число – количество клеток, из которых робот сможет выбраться до вертолетной площадки (если верить, что он не развалится по дороге).

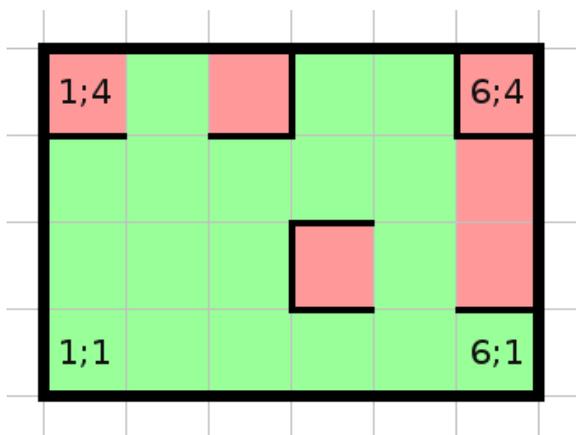


Примеры

stdin	stdout
6 4 3 4 2 4 4 6 4 6 4 2 6 4 4 3 3 4 6 2 1 4 3 push_down(); push_left(); push_down();	18

Пояснение

Карта из первого примера. Зеленым цветом обозначены клетки, из которых робот сможет выбраться, красным — из которых не сможет.



Так как робот не умеет останавливаться во время выполнения программы, то в клетке (1;1) он должен находиться после выполнения программы. Если он придет в эту клетку, потом уйдет из нее и не вернется, то выбраться из подземелья ему не удастся.

Тесты к этой задаче состоят из пяти групп:

- Тест 1. Тест из условия. Оценивается в 0 баллов.
- Тесты 2 – 7. В тестах этой группы $1 \leq n, m \leq 10$, $k \leq 30$. Оцениваются в 30 баллов.
- Тесты 8 – 13. В тестах этой группы $1 \leq n, m \leq 100$, $k \leq 500$. Оцениваются в 30 баллов.
- Тесты 14 – 21. В тестах этой группы дополнительные ограничения отсутствуют. Оцениваются в 40 баллов.