

Задача 1. Папины дочки

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Но наш герой не сдаётся. Слезы утри!

группа Uma2rman

Сегодня 8 марта, с чем мы поздравляем прекрасную половину человечества! В то время, пока вы пишете олимпиаду, Сергей Васнецов находится в торговом центре и выбирает подарки своим дочерям. Это задача непростая, ведь дочерей у него целых N ! (*это восклицание, а не факториал*)

Сергей решил подарить своим дочкам конфеты. Каждой дочке — по одной подарочной коробке. Но в торговом центре большой выбор самых разнообразных коробок с разным количеством конфет. Но даже это — не самая большая проблема.

Всю неделю дочки подходили к папе и рассказывали про свои *требования* к подарку. Все требования звучат схожим образом: «В моем подарке конфет должно быть **не меньше**, чем у ...» (*далее следует имя одной из сестер*)

К счастью, Сергей хорошо знает своих дочек и особенности их взаимоотношений, поэтому он уверен, что какую бы пару дочек он ни взял, то, исходя из всех требований, для этой пары будет гарантированно верно **хотя бы одно** из двух утверждений:

- у первой дочки должно быть не меньше конфет, чем у второй;
- у второй дочки должно быть не меньше конфет, чем у первой.

Имея под рукой набор требований, а также зная ассортимент торгового центра, Сергею нужно понять, сможет ли он обрадовать всех своих дочерей. У него под рукой нет компьютера, поэтому он обратился за помощью к вам.

Формат входных данных

В первой строке входных данных записаны два целых числа N и M — количество дочерей в семье Васнецовых и количество требований, которые они сообщили отцу ($2 \leq N \leq 10^5$, $1 \leq M \leq 2 \cdot 10^5$).

В следующих M строках перечислены требования, по одному на строке. Требование состоит из двух целых чисел x и y , записанных через пробел, которые обозначают, что дочка x хочет, чтобы в ее подарке было не меньше конфет, чем у дочки y ($1 \leq x, y \leq N$, $x \neq y$). Все требования различны.

В следующей строке записано целое число K — количество коробок конфет, которые есть в торговом центре ($N \leq K \leq 2 \cdot 10^5$).

В последней строке через пробел записаны K целых чисел — количество конфет в каждой коробке. Коробка не может быть пустой. Количество конфет в коробке не превосходит 10^9 . В некоторых коробках количество конфет может быть одинаковым.

Формат выходных данных

В первой строке выведите YES, если Сергею Васнецову удастся обрадовать всех своих дочек, или NO, если все требования выполнить не удастся.

В случае положительного ответа, во второй строке выведите N чисел через пробел — количество конфет, которое получит каждая дочка. Порядок дочерей в ответе должен совпадать с порядком во входных данных.

Система оценки

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

Подзадача	Баллы	Ограничения	Необходимые подзадачи
1	0	Тесты из условия	
2	12	$N \leq 500, M \leq 1000$ Нет коробок с одинаковым числом конфет	
3	36	$N \leq 500, M \leq 1000$	1, 2
4	12	Нет коробок с одинаковым числом конфет	2
5	40	Без дополнительных ограничений	1, 2, 3, 4

Примеры

стандартный ввод	стандартный вывод
5 6 1 2 1 4 5 1 4 5 3 5 3 4 5 2 3 2 1 2	YES 2 1 3 2 2
2 2 1 2 2 1 2 1 2	NO

Задача 2. Антошка, Антошка, пойдём копать картошку

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Это мы не проходили, это нам не давали.

Па-рам-пам-пам, па-рам-пам-пам!

— Антошка

Сегодня Антошке предстоит копать картошку. Кусты картошки, которые ему нужно выкопать, расположены в один ряд. Также для каждого куста известно, сколько клубней картофеля в нем находится. Так как Антошка все-таки прошел четность чисел, то он придумал следующее «правило копания»:

- Антошка начинает копать картошку с первого куста и в любом случае выкапывает его.
- Если количество клубней, которое он выкопал из очередного куста, чётно, то он пропустит следующий куст и сразу начнет копать куст, расположенный через один от последнего выкопанного. Число ноль считается чётным.
- Если же количество клубней, которое он выкопал из очередного куста, нечётно, то Антошка просто перейдет к выкапыванию следующего куста.
- Если куста, к которому следует перейти, нет (ряд подошел к концу), Антошка заканчивает копать картошку.

Однако вы не хотите, чтобы Антошка из-за своего выдуманного правила собрал мало картошки, поэтому до того как Антошка начнет копать, вам разрешается убрать из некоторых (в том числе, из всех или из никаких) кустов некоторое количество клубней. Разрешено оставлять куст, в котором нет ни одного клубня.

Вам требуется определить максимальное количество клубней картошки, которое выкопает Антошка, следуя своему правилу, после ваших действий. Клубни, которые вы забрали до того, как он начал копать, не идут в счет Антошке.

Формат входных данных

Первая строка содержит число N — количество кустов картошки в ряду ($1 \leq N \leq 10^5$).

Во второй строке записано N целых чисел a_i — количество клубней картошки в i -м кусте ($1 \leq a_i \leq 10^9$).

Формат выходных данных

Требуется вывести одно число — максимальное количество клубней картошки, которое выкопает Антошка.

Примеры

стандартный ввод	стандартный вывод
5 3 4 1 2 3	11

Система оценки

В этой задаче 20 скрытых тестов, каждый тест, кроме примера из условия, оценивается в 5 баллов.

Замечание

В тесте из условия вам достаточно достать один клубень из куста под номером «4». Таким образом, Антошка выкопает кусты в следующем порядке: $1 \rightarrow 2 \rightarrow 4 \rightarrow 5$ и соберет 11 клубней (3 из первого куста, 4 из второго, 1 из четвертого и 3 из пятого).

Задача 3. Шахматный турнир

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Петя любит шахматы и внимательно следит за всеми шахматными турнирами, в которых участвуют его любимые игроки. Прямо сейчас как раз проходит большой турнир, в котором принимают участие целых 2^N гроссмейстеров!

Турнир проходит по системе с выбыванием после двух поражений. Это значит, что турнирная сетка делится на две части: верхнюю и нижнюю. В верхней сетке игроки играют по стандартной нокаут-системе — после каждого тура проигравшие игроки из нее выбывают. Но при этом они не выбывают из турнира, а оказываются в нижней сетке и продолжают играть в ней. Только после первого поражения в нижней сетке участник окончательно выбывает из турнира.

Если после очередного тура из верхней сетки вышло k игроков, а в нижней сетке все еще больше, чем k игроков, то выбывшие из верхней сетки ожидают несколько туров, пока в нижней сетке не останется ровно k игроков. После этого в очередном туре нижней сетки проигравший в i -й игре верхней сетки играет с победителем i -й игры в нижней сетке.

Во всех остальных случаях игроки очередного тура распределяются стандартным образом внутри каждой сетки: в i -й игре очередного тура играют победители $(2i - 1)$ -й и $(2i)$ -й игры предыдущего тура. Аналогичным образом формируются пары первого тура нижней сетки: в i -й игре первого тура нижней сетки играют проигравшие $(2i - 1)$ -й и $(2i)$ -й игры первого тура верхней сетки.

Перед последним туром в каждой сетке остается один игрок, между этими двумя игроками проводится финальный тур.

Каждый день в каждой сетке проводится максимум один тур. Игры в верхней сетке начинаются с первого дня и далее идут без перерывов, пока в ней не определится победитель. После этого победитель верхней сетки ожидает, когда определится победитель нижней сетки. Игры в нижней сетке начинаются со второго дня и далее идут без перерывов. Финальный тур проводится на следующий день после последнего тура нижней сетки.

Посмотрите замечание к примеру из условия, чтобы подробнее разобраться с расписанием турнира.

Петя знает, что в первом туре все участники распределены по парам по убыванию рейтинга ($\text{№}1$ играет с $\text{№}2$, $\text{№}3$ играет с $\text{№}4$, и так далее, $\text{№}(2^N - 1)$ играет с $\text{№}(2^N)$). Также Петя знает, что когда встречаются два игрока, всегда побеждает тот игрок, у которого рейтинг больше. То есть всегда побеждает игрок с меньшим номером.

Петя хочет следить за турниром не каждый день, а только в те дни, когда играют его любимые игроки. А если быть точнее, его интересуют конкретные пары игроков, список которых он Вам любезно сообщил. Для каждой пары он хочет знать, сыграют ли эти гроссмейстеры друг против друга на этом турнире, и если да, то в какие дни.

Формат входных данных

В первой строке записано целое число N — двоичный логарифм от количества участников турнира ($2 \leq N \leq 60$).

Во второй строке записано целое число Q — количество запросов ($1 \leq Q \leq 2 \cdot 10^4$).

В следующих Q строках даны запросы, по одному на строке. Каждый запрос состоит из двух чисел x, y — номеров интересующих игроков, записанных через пробел ($1 \leq x < y \leq 2^N$).

Формат выходных данных

Для каждого запроса в отдельной строке нужно вывести число K — сколько раз сыграют друг с другом два игрока из запроса, и далее K чисел — номера дней, в которые эти игроки будут играть, в порядке возрастания.

Система оценки

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

Подзадача	Баллы	Ограничения	Необходимые подзадачи
1	0	Тесты из условия	
2	32	$N \leq 20$	1
3	22	$N \leq 30$	1, 2
4	46	$N \leq 60$	1, 2, 3

Примеры

стандартный ввод	стандартный вывод
2	2 1 4
6	1 2
1 2	0
1 3	1 3
1 4	1 2
2 3	1 1
2 4	
3 4	
4	2 1 8
5	1 3
1 2	1 6
2 3	1 7
2 10	0
2 9	
2 8	

Пояснение к примеру

На следующей странице приведено расписание игр для турнира из 16 участников (пример №2). Серым цветом выделены игры, которые проходят в нижней сетке.

Обратите внимание, что после 3-го дня игроки №5 и №13, проиграв в верхней сетке, пропустят 4-й день и начнут играть в нижней сетке только на 5-й день, когда в нижней сетке останется ровно 2 игрока. Аналогично, игрок №9 начнет играть в нижней сетке только на 7-й день.

Всесибирская открытая олимпиада школьников по информатике
Заключительный этап, 7-11 классы, 8 марта 2025 г.

День 1	1 2	3 4	5 6	7 8	9 10	11 12	13 14	15 16
День 2	1 3	5 7	9 11	13 15	2 4	6 8	10 12	14 16
День 3	1 5	9 13			2 3	6 7	10 11	14 15
День 4	1 9				2 6	10 14		
День 5					2 5	10 13		
День 6					2 10			
День 7					2 9			
День 8	1 2							

Задача 4. Прогулка в парке

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Акакий и Иннокентий после уроков зашли в парк. Сегодня на факультативе по математике они узнали, что мекс множества чисел — это наименьшее неотрицательное целое число, которое отсутствует в множестве. Например, мекс множества $\{0, 1, 2, 3, 5\}$ равен 4, а мекс множества $\{1, 2\}$ равен 0.

Новое понятие не выходило из головы, и во время прогулки Акакий предложил следующую задачу: нужно построить множество неотрицательных целых чисел, сумма всех элементов которого равна некоторому числу K , так, чтобы произведение максимального элемента множества на мекс множества, было наибольшим. Иннокентий сходу придумал решение, но боится, что мог где-то ошибиться. Помогите ему посчитать искомые значения для самопроверки!

Формат входных данных

В первой строке входных данных дано одно число T — количество тестов ($1 \leq T \leq 10^5$).

В следующих T строках записаны необходимые суммы элементов множеств K в каждом тесте ($1 \leq K \leq 10^{12}$).

Формат выходных данных

Для каждого теста выведите единственное число — наибольшее значение произведения мекса и максимального элемента множества.

Система оценки

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

Подзадача	Баллы	Ограничения	Необходимые подзадачи
1	0	Тесты из условия	
2	21	$K \leq 50$	1
3	23	$K \leq 10^6$	1, 2
4	27	$K \leq 10^9$	1, 2, 3
5	29	$K \leq 10^{12}$	1, 2, 3, 4

Пример

стандартный ввод	стандартный вывод
2	6
3	6
4	

Пояснение к примеру

В первом примере максимально значение можно получить, используя, например, множество $\{0, 1, 2\}$, максимальное значение в нём равно 2, а мекс — 3.

Задача 5. Стол и вода

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	256 мегабайт

В один прекрасный весенний день Дима стал решать очередную интересную задачу. Задача была непростая, и через некоторое время у него накопилось много листов бумаги с промежуточными вычислениями, которые оказались хаотично разбросаны по столу (возможно, полностью или частично перекрывая друг друга). К несчастью, на столе стоял стакан воды, который Дима случайно опрокинул, и вода начала распространяться равномерно во всех направлениях со скоростью **1 сантиметр в секунду**. По листу вода распространяется с такой же скоростью и так же равномерно.

Дима задумался о том, сколько листов будет испорчено, если он не поторопится остановить распространение жидкости по столу. Лист считается испорченным, если вода хотя бы коснулась его. Для упрощения решения этой задачи Дима наложил на стол координатную плоскость, причем так, что расстояние между парой соседних делений координатных прямых этой плоскости, равно одному сантиметру.

Помогите Диме поскорее посчитать ущерб!

Формат входных данных

В первой строке входных данных дана пара целых чисел x_w, y_w — координаты стакана воды ($|x_w|, |y_w| \leq 10^9$). Во второй строке задано число N — количество листов на столе ($1 \leq N \leq 10^5$). В каждой из следующих N строк заданы восемь целых чисел $x_{i,1}, y_{i,1}, x_{i,2}, y_{i,2}, x_{i,3}, y_{i,3}, x_{i,4}, y_{i,4}$, задающих координаты четырёх вершин листа ($|x_{i,k}|, |y_{i,k}| \leq 10^9$). Гарантируется, что точки задают прямоугольник, и они заданы в порядке обхода (по часовой или против часовой стрелки).

В следующей строке задано число Q — количество вопросов Димы ($1 \leq Q \leq 10^5$). В каждой из следующих Q строк задано целое число t_i , определяющее интересующий его момент времени (считая от момента опрокидывания стакана) ($0 \leq t_i \leq 10^9$).

Формат выходных данных

Для каждого t_i нужно вывести количество листов, испорченных водой через t_i секунд после опрокидывания стакана.

Система оценки

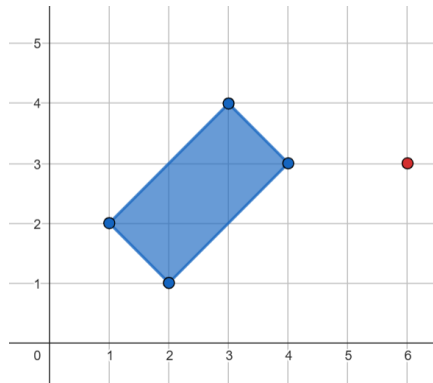
Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Ограничения	Необходимые подзадачи
1	0	Тесты из условия	
2	27	$1 \leq N, Q \leq 1000$	1
3	32	$1 \leq N, Q \leq 10^5$ Все листы размещены параллельно осям координат	
4	41	$1 \leq N, Q \leq 10^5$	1, 2, 3

Пример

стандартный ввод	стандартный вывод
6 3	1
1	0
1 2 3 4 4 3 2 1	
2	
2	
0	

Пояснение к примеру



На рисунке представлен тест из условия: начальное положение стакана задано точкой с координатами $(6, 3)$ а расположение единственного листа — заштрихованным прямоугольником. Можно заметить, что через две секунды после опрокидывания стакана, вода достигнет листа.

Задача 6. Эксперимент

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайт



В Новосибирском Академгородке около Института Цитологии и Генетики поставлен памятник лабораторной мыши, которая плетет ДНК. Он поставлен в благодарность за то, что эксперименты, проведенные над этими грызунами, привели к научным открытиям, позволившим создать новые лекарства, которые в итоге спасли жизнь многим людям.

Сегодня ученым требуется провести еще один эксперимент. Для наблюдения за мышами они построили стеклянный лабиринт, состоящий из комнат-норок, где живут мышки, и проходов между норками. Лабиринт устроен так, что *из любой норки по проходам мышка может попасть в любую другую норку, причем единственным способом*. Назовем расстоянием между двумя норками количество проходов, которые должна преодолеть мышь, чтобы добраться из одной норки в другую.

Лабиринт расположили в большом ангаре, где зимой бывает очень холодно — Сибирь, всё-таки. Для того, чтобы мыши не замерзли, нужно было найти способ согреть их.

Согреть мышей решили обогревателями, располагая их в некоторых норках. Для работы в этом лабиринте обычные обогреватели не подойдут, проводить электричество в норы очень затратно и небезопасно для мышей, поэтому ученые придумали использовать новшество — автономные обогреватели, которые для работы не требуют электричества, но имеют другой недостаток — они не только греют, но и охлаждают.

Для того, чтобы удостовериться в качественном обогреве лабиринта, необходимо провести некоторые расчеты. С этой целью для каждой норки была введена характеристика, которую назвали уровнем теплоты. Изначально уровень теплоты каждой норки равен 0.

Каждый обогреватель имеет характеристику — мощность x . Когда он устанавливается в какую-либо норку v , уровни теплоты во **ВСЕХ** норках лабиринта изменяются следующим образом:

- В норе v уровень теплоты увеличивается на мощность обогревателя x .
- Во всех норках на расстоянии 1 от v уровень теплоты увеличивается на $x - 1$.
- Во всех норках, расположенных на расстоянии 2 от v , уровень теплоты увеличивается на $x - 2$.
- И так далее...

Соответственно, в норках, далеких от v , уровень теплоты, возможно, даже уменьшится. В одну норку ученые могут установить несколько обогревателей.

Установка лабиринта и обогревателей — дело хлопотное, требует времени и сил. Чтобы время зря не пропадало, и ни одна лабораторная мышь не пострадала, ученые просят Вас помочь провести предварительные расчеты.

Для заданного лабиринта требуется обработать серию запросов двух видов:

- узнать текущий уровень теплоты в некоторой норке,
- разместить в определенной норке обогреватель заданной мощности.

Формат входных данных

Первая строка входных данных содержит целое число N — количество норок в лабиринте ($1 \leq N \leq 2 \cdot 10^5$). Норки пронумерованы числами от 1 до N .

В следующих $N - 1$ строках записано по два целых числа u и v , которые означают, что между норками с номерами u и v есть проход ($1 \leq u, v \leq N$).

Следующая строка содержит целое число Q — количество запросов, которые нужно обработать ($1 \leq Q \leq 2 \cdot 10^5$).

Далее следует Q строк, в каждой строке записано по одному запросу. Запросы имеют следующий формат:

1. ? v — узнать уровень теплоты в норке v ($1 \leq v \leq N$).
2. + v x — разместить обогреватель мощностью x в норке v ($1 \leq v \leq N, 1 \leq x \leq 10^9$).

Формат выходных данных

Для каждого запроса первого вида нужно вывести ответ на этот запрос в отдельной строке, в том же порядке, в каком они встречаются во входных данных.

Система оценки

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

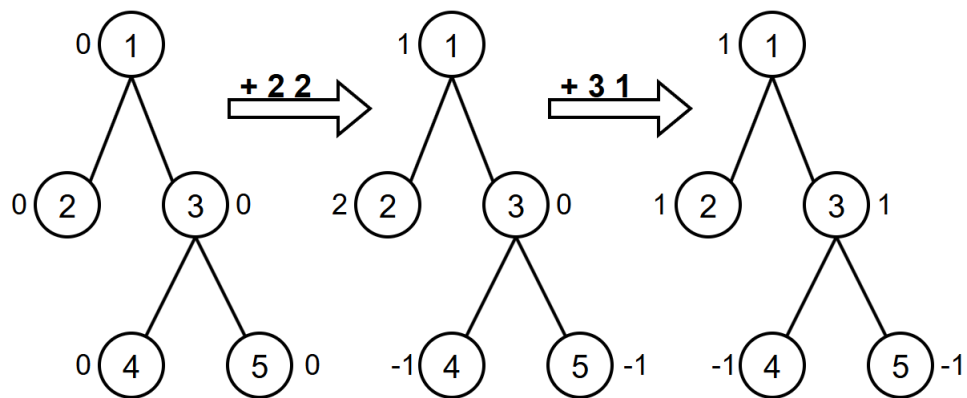
Подзадача	Баллы	Ограничения	Необходимые подзадачи
1	0	Тесты из условия	
2	7	$1 \leq N, Q \leq 10000$	1
3	8	$1 \leq N, Q \leq 5 \cdot 10^4$ Существует норка, напрямую соединенная со всеми остальными	
4	16	$1 \leq N, Q \leq 5 \cdot 10^4$ Из каждой норки выходит не более двух проходов	
5	21	$1 \leq N, Q \leq 5 \cdot 10^4$	1, 2, 3, 4
6	11	$1 \leq N \leq 10^5, 1 \leq Q \leq 5 \cdot 10^4$	1, 2, 3, 4, 5
7	24	$1 \leq N \leq 2 \cdot 10^5, 1 \leq Q \leq 10^5$	1, 2, 3, 4, 5, 6
8	13	$1 \leq N, Q \leq 2 \cdot 10^5$	1, 2, 3, 4, 5, 6, 7

Пример

стандартный ввод	стандартный вывод
5	0
1 2	-1
1 3	2
3 4	1
3 5	1
9	1
? 3	-1
+ 2 2	
? 4	
? 2	
+ 3 1	
? 1	
? 2	
? 3	
? 5	

Пояснение к примеру

На рисунке ниже показана схема лабиринта: норки обозначены окружностями с номерами, проходы – отрезками, соединяющими некоторые из норок. Около каждой норки указаны уровни теплоты: на левой схеме — в самом начале, и далее, после каждого запроса модификации.



Задача 7. Тестирование уровней

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Аркадий — независимый разработчик игр.

В его новом проекте игрок управляет космическим кораблём, начинающим свой путь в точке $(0, 0)$ и движущимся с постоянной скоростью вдоль оси Ox , при этом на него действует постоянное ускорение вдоль оси Oy . Игрок может изменять направление ускорения (умножать его значение на -1), при этом его текущая скорость вдоль оси Oy становится равной 0 . Цель игрока — пройти через все врата на пути. Врата — это отрезки, параллельные оси Oy . При прохождении через врата вертикальная скорость космического корабля также становится равной 0 .

Аркадий хочет, чтобы уровни, которые он создаёт, можно было пройти *идеально*. При идеальном прохождении изменять направление гравитации можно

- в момент прохождения врат;
- между вратами, но не более одного раза между каждой парой соседних врат.

Помогите ему протестировать уровни!

Формат входных данных

В первой строке входных данных задано три целых числа N, V_x, A_y — количество врат, скорость вдоль оси Ox и модуль начального ускорения вдоль оси Oy ($1 \leq N \leq 10^5$, $1 \leq V_x \leq 10^6$, $1 \leq A_y \leq 100$). **Изначально ускорение отрицательно.**

В следующих N строках заданы описания врат. Каждая строка содержит три целых числа: x, y_1, y_2 — позиция врат по оси Ox и y -координаты концов отрезка ($1 \leq x \leq 10^9$, $-10^9 \leq y_1 \leq y_2 \leq 10^9$).

Гарантируется, что $x_i < x_{i+1}$.

Формат выходных данных

В первой строке выведите YES или NO, в зависимости от того, можно пройти уровень идеально или нет.

В случае, если уровень проходится идеально — выведите количество изменений гравитации, а затем время изменений гравитации в возрастающем порядке.

Абсолютная или относительная погрешность y -координат точек пересечения врат в ваших прохождениях не должна превышать 10^{-4} .

Примеры

стандартный ввод	стандартный вывод
2 1 1	YES
2 -2 -2	1
6 6 6	2
3 1 1	YES
1 -1 1	2
2 -1 1	1
3 -1 1	1.5
1 1 1	NO
1 -1000 -999	

Пояснение к примеру

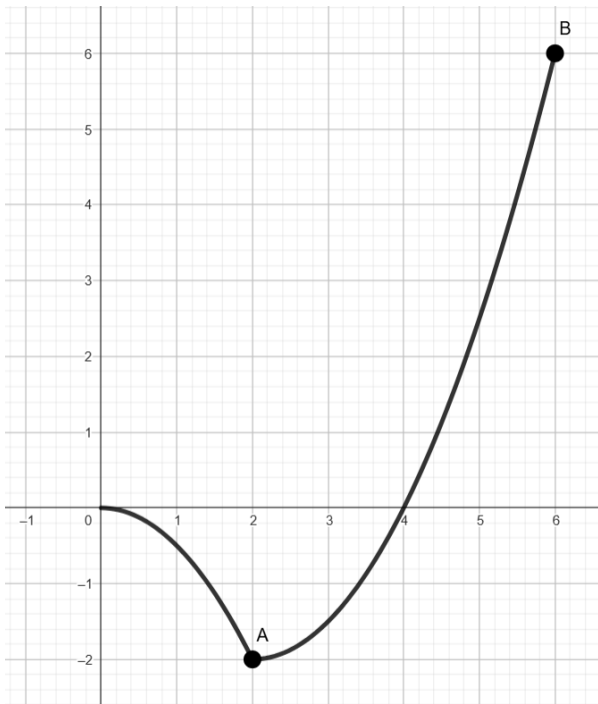


Рис. 1: Иллюстрация первого примера

В первом примере единственный возможный путь проходит через точки $(0, 0)$, $(2, -2)$ и $(6, 6)$, при этом гравитацию нужно изменить в момент времени 2.

Во втором примере показаны 2 способа пройти идеально уровень: выбрав путь **P** или путь **Q**. В первом случае нужно изменить направление гравитации в моменты времени 1 и $\frac{3}{2}$. Во втором — в моменты времени 0 и 2. Обратите внимание, что это **не единственные** способы идеально пройти уровень.

Система оценки

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

Подзадача	Баллы	Ограничения	Необходимые подзадачи
1	0	Тесты из условия	
2	5	$1 \leq N \leq 10$, все врата являются точками	1
3	13	$1 \leq N \leq 1000$, все врата являются точками	1, 2
4	21	$1 \leq N \leq 2 \cdot 10^5$, все врата являются точками	1, 2, 3
5	7	$1 \leq N \leq 10$	1, 2
6	20	$1 \leq N \leq 1000$	1, 2, 3, 5
7	34	$1 \leq N \leq 10^5$	1, 2, 3, 4, 5, 6

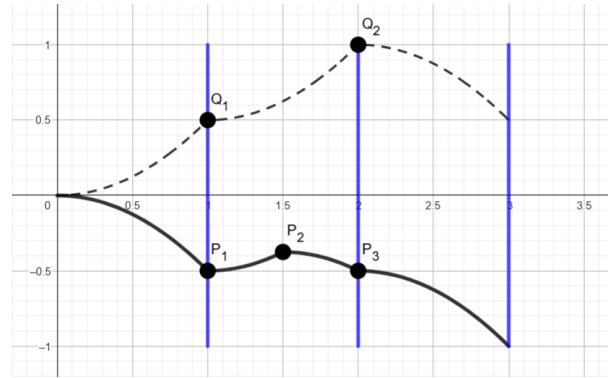


Рис. 2: Иллюстрация второго примера