Задача 1. B2B AI SaaS Six-Seven Startup

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

Вася придумал идею для B2B AI SaaS Six-Seven стартапа: бот в мессенджере, который любое число превращает в 67.

На каждом шаге превращения бот может применять к текущему числу x одну из следующих операций:

1. $x \to x + 67$,
2. $x \to |67 - x|$ (модуль разности),
3. $x \to x \bmod 67$ (остаток от деления),
4. $x \to \lfloor x/67 \rfloor$ (целочисленное деление).

Вася хочет тратить как можно меньше вычислительных ресурсов, поэтому превращение в 67 должно происходить за минимально возможное количество операций.

У его AI-агента закончились токены, поэтому он просит вас посчитать минимальное количество операций, необходимых для превращения заданного числа в 67.

Формат входных данных

В первой строке вводится число Q — количество запросов ($1 \leqslant Q \leqslant 10^5$).

Далее следует Q строк, в каждой записано единственное неотрицательное целое число x, которое нужно превратить в 67 ($0 \leqslant x \leqslant 676767$).

Формат выходных данных

Необходимо вывести Q строк, по одному числу в строке: минимальное количество операций для каждого из Q запросов.

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	10	все $x \leqslant 676$	1
3	20	все $x \leqslant 6767$	1, 2
4	30	все $x \leqslant 67676$	1, 2, 3
5	40	все $x \leqslant 676767$	1, 2, 3, 4

Пример

стандартный ввод	стандартный вывод
3 0 6 67	1 2 0

Задача 2. Магические кристаллы

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

К первому занятию по алхимии ученики одной известной школы магии получили книгу Николаса Фламеля — знаменитого создателя философского камня. Многие надеялись узнать из неё секрет этого удивительного артефакта и когда-нибудь повторить достижение автора.

На занятии преподаватель объяснил, что философский камень — особая разновидность магических кристаллов. Каждый магический кристалл хранит энергию, количество которой называют его **зарядом** и измеряют целым числом условных единиц. С помощью заклинаний кристаллы можно объединять, получая новые кристаллы с другим зарядом. Для создания философского камня требуется сложная последовательность заклинаний, и ученикам ещё многое предстоит освоить.

Начать обучение преподаватель предложил с упражнения на два простейших заклинания — «*Максимус*» и «*Минимус*». Для этого он выложил на стол ряд из N кристаллов, заряды которых заданы массивом a. Каждое из заклинаний объединяет **два соседних в ряду** кристалла в один:

- при использовании заклинания «*Максимус*» новый кристалл получает больший из зарядов выбранных кристаллов;
- при использовании заклинания «*Минимус*» новый кристалл получает меньший из зарядов выбранных кристаллов.

Получившийся кристалл занимает место выбранной пары, а остальные сохраняют свой взаимный порядок. После каждого применения заклинания общее количество кристаллов уменьшается на один.

Затем преподаватель передал ученикам свиток со списком из $N-1$ заклинаний. Их необходимо выполнить **строго в порядке**, в котором они записаны в свитке. Для каждого заклинания ученики могут самостоятельно выбрать пару соседних кристаллов, к которой его применят. В исходном упражнении заклинания могли идти в произвольном порядке. К счастью, преподаватель подготовил для новичков учебный свиток, чтобы те реже путали заклинания. В нём сначала идут X заклинаний «*Максимус*», а затем — $N-1-X$ заклинаний «*Минимус*». Таким образом, после применения всех заклинаний останется ровно один кристалл. Чем больше его заряд, тем выше оценка за занятие. Ученики хотят получить наивысшую оценку и готовы пустить в ход все свои знания и магические способности!

Определите наибольший заряд, который ученики могут получить у оставшегося кристалла, оптимально выбирая пары соседних кристаллов для каждого заклинания.

Формат входных данных

Первая строка содержит два целых числа N и X — количество кристаллов и количество заклинаний «*Максимус*» в свитке ($1 \leqslant N \leqslant 2 \cdot 10^5$, $0 \leqslant X \leqslant N-1$).

Вторая строка содержит N целых чисел $a_1, a_2, \ldots, a_N$ — заряды кристаллов в порядке их расположения ($1 \leqslant a_i \leqslant 10^9$).

Формат выходных данных

Выведите одно целое число — наибольший возможный заряд кристалла, который останется после применения всех заклинаний.

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	9	$N \leqslant 2 \cdot 10^5$, $X = 0$ или $X = N - 1$	—
3	13	$N \leqslant 10$	1
4	17	$N \leqslant 200$	1, 3
5	17	$N \leqslant 2000$	1, 3, 4
6	44	$N \leqslant 2 \cdot 10^5$	1, 2, 3, 4, 5

Пример

стандартный ввод	стандартный вывод
5 2 4 1 7 2 6	4

Замечание

В примере можно применить заклинания следующим образом:

1. применить *Максимус* к соседним кристаллам с зарядами 4 и 1, получив массив зарядов $[4, 7, 2, 6]$;
2. применить *Максимус* к соседним кристаллам с зарядами 7 и 2, получив массив зарядов $[4, 7, 6]$;
3. применить *Минимус* к соседним кристаллам с зарядами 4 и 7, получив массив зарядов $[4, 6]$;
4. применить *Минимус* к оставшимся кристаллам, получив кристалл с зарядом 4.

Получить ответ больше 4 в этом примере невозможно.

Задача 3. Опять все перепутали...

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

К началу олимпиадного сезона 2026–2027 организаторы подготовили расписание соревнований. В него вошли N соревнований: часть из них состоится в 2026 году, остальные — в 2027-м. При переносе расписания на сайт произошёл сбой: у некоторых соревнований изменился год проведения. Соревнование, запланированное на 2026 год, могло оказаться в 2027-м, и наоборот. К счастью, у организаторов сохранилось исходное расписание, с которым можно сверить данные сайта. Поэтому все еще возможно исправить такое недоразумение и привести расписание на сайте в корректное состояние.

Поскольку первые три цифры года одинаковые, список соревнований можно представить в виде строки длины N, записав для каждого соревнования последнюю цифру года его проведения. Строка A соответствует расписанию на сайте, а строка B — исходному расписанию. Обе строки состоят только из символов «6» и «7». Соревнования нумеруются в порядке следования в расписании, начиная с единицы. Обратите внимание, что в исходном расписании соревнования не обязательно отсортированы по году проведения.

Для исправлений доступен старый редактор массовых изменений. За одну операцию он позволяет выбрать ровно 6 или ровно 7 соревнований и поменять год у каждого из них: 2026 на 2027, а 2027 на 2026. В пределах одной операции все выбранные соревнования должны быть различными, но в последующих операциях разрешается снова выбирать соревнования, которые уже изменялись ранее. Иными словами, во всех выбранных позициях строки A символ «6» заменяется на «7», а символ «7» — на «6».

Чтобы исправления появились в расписании на сайте, редактор отправляет каждую команду на сервер отдельным запросом. Обработка каждого запроса занимает довольно много времени. Так как времени до начала соревнований остается совсем немного, организаторы просят вас помочь исправить расписание до открытия регистрации: определите, какие операции и в каком порядке нужно выполнить, чтобы превратить строку A в строку B за минимальное количество операций. Выведите это количество и сами операции. Если получить требуемую строку невозможно, выведите число -1.

Формат входных данных

Первая строка содержит одно целое число N — длину строк ($1 \leqslant N \leqslant 2 \cdot 10^5$).

Во второй строке записана строка A длины N — последние цифры годов проведения соревнований в расписании на сайте.

В третьей строке записана строка B длины N — последние цифры годов проведения соревнований в исходном расписании.

Обе строки состоят только из символов «6» и «7».

Формат выходных данных

Если превратить строку A в строку B невозможно, выведите единственное число -1.

Иначе в первой строке выведите целое число M — минимальное количество операций.

В каждой из следующих M строк опишите одну операцию в порядке выполнения. Сначала выведите число K ($K = 6$ или $K = 7$), затем K различных целых чисел $p_1, p_2, \ldots, p_K$ — выбранные позиции ($1 \leqslant p_i \leqslant N$).

Если существует несколько последовательностей из минимального количества операций, выведите любую из них.

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	13	$N \leqslant 7$	1
3	8	$N \leqslant 8$	1, 2
4	17	$N \leqslant 15$	1, 2, 3
5	29	$N \leqslant 2000$	1, 2, 3, 4
6	33	$N \leqslant 2 \cdot 10^5$	1, 2, 3, 4, 5

Примеры

стандартный ввод	стандартный вывод
7 6666666 7666666	2 6 7 6 5 4 3 2 7 2 3 4 5 6 7 1
5 66666 76666	-1
2 67 67	0

Замечание

В первом примере сначала можно изменить год у соревнований с номерами от 2 до 7 и получить строку 6777777. Затем изменить год у всех семи соревнований и получить искомую строку. Одной операции точно недостаточно: строки отличаются только в одной позиции, а каждая операция изменяет ровно шесть или семь позиций.

Во втором примере соревнований всего пять, поэтому нельзя выполнить ни одной операции. Начальная строка отличается от требуемой, следовательно, ответ равен -1.

Задача 4. Странный калькулятор

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

В лаборатории старых вычислительных устройств обнаружили необычный калькулятор. Сразу после включения на его экране отображается число 1. На панели калькулятора расположены кнопки с цифрами от 2 до 9. При нажатии кнопки с цифрой d калькулятор умножает число на экране на d и показывает результат.

Исследователи решили изучить все возможные последовательности нажатий. Каждую кнопку можно нажимать сколько угодно раз, а порядок нажатий имеет значение для исследователей. Две последовательности считаются различными, если они отличаются хотя бы одним нажатием, даже если в результате на экране появляется одно и то же число.

Последовательность может состоять и из нуля нажатий. В этом случае число на экране останется равным 1.

Исследователям предстоит обработать q запросов. Каждый запрос задаётся двумя целыми числами l и r. Для каждого запроса необходимо определить количество последовательностей нажатий, после которых на экране окажется число от l до r включительно.

Поскольку ответ может быть очень большим, выведите его по модулю $10^9 + 7$.

Формат входных данных

В первой строке задано целое число q — количество запросов($1 \leqslant q \leqslant 2 \cdot 10^5$).

В следующих q строках заданы запросы. Каждая строка содержит два целых числа l и r ($1 \leqslant l \leqslant r \leqslant 10^{18}$).

Формат выходных данных

Для каждого запроса выведите в отдельной строке одно целое число — количество подходящих последовательностей нажатий по модулю $10^9 + 7$.

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	6	$r \leqslant 10^6$	1
3	17	$\sum (r - l) \leqslant 10^6$	1
4	20	$r \leqslant 10^9$	1, 2
5	12	$q \leqslant 10$	1
6	45	Нет дополнительных ограничений	1, 2, 3, 4, 5

Пример

стандартный ввод	стандартный вывод
4 1 1 5 6 20 20 1 24	1 4 5 67

Задача 5. Нормы гроссмейстера

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

Василий — опытный шахматист, который уже много лет мечтает стать гроссмейстером. Недавно он наконец принял участие в долгожданном турнире, в котором сыграл n партий.

Для каждой партии известны рейтинг соперника r_i и результат партии: W, если Василий одержал победу, D, если партия завершилась вничью, и L, если Василий проиграл.

Международная шахматная федерация предъявляет строгие требования к получению нормы гроссмейстера. Чтобы кандидат мог рассчитывать на норму, его выступление должно соответствовать нескольким условиям.

Во-первых, заявка должна представлять собой **непрерывный отрезок** сыгранных партий (последовательность нескольких подряд сыгранных партий). При этом количество партий в заявке должно быть не меньше $k_{\min}$ и не больше $k_{\max}$.

Во-вторых, чтобы исключить возможность получения нормы за счёт исключительно слабых соперников, средний рейтинг соперников на выбранном отрезке должен быть не меньше R_{avg}.

Наконец, кандидат должен набрать не менее p процентов от максимально возможного количества очков на этом отрезке. За победу начисляется 1 очко, за ничью — 0.5 очка, а за поражение — 0 очков. Таким образом, на отрезке из k партий кандидат должен набрать не менее $\frac{p \cdot k}{100}$ очков.

Кроме того, любые две заявки Василия на получение нормы не должны пересекаться по партиям: каждая партия может входить не более чем в одну выбранную заявку.

Василий устал после турнира и просит вас помочь ему определить максимальное количество норм гроссмейстера, которые он может получить.

Формат входных данных

Первая строка содержит целое число t — количество наборов входных данных ($1 \leqslant t \leqslant 10^5$).

Каждый набор входных данных состоит из трёх строк.

Первая строка содержит пять целых чисел n, $k_{\min}$, $k_{\max}$, R_{avg} и p ($1 \leqslant n \leqslant 2 \cdot 10^5$, $1 \leqslant k_{\min} \leqslant k_{\max} \leqslant n$, $0 \leqslant R_{\text{avg}} \leqslant 4000$, $1 \leqslant p \leqslant 100$).

Вторая строка содержит n целых чисел $r_1, r_2, \ldots, r_n$ ($0 \leqslant r_i \leqslant 4000$), где r_i — рейтинг соперника в i-й партии.

Третья строка содержит строку s длины n, состоящую только из символов W, D и L. Символ s_i обозначает результат Василия в i-й партии.

Гарантируется, что сумма n по всем наборам входных данных не превосходит $2 \cdot 10^5$.

Формат выходных данных

Для каждого набора входных данных выведите одно целое число — максимальное количество непересекающихся по партиям отрезков, каждый из которых удовлетворяет всем требованиям федерации.

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	6	$\sum n \leqslant 500$	1
3	7	$\sum n \leqslant 5000$	1, 2
4	10	$k_{\max} - k_{\min} \leqslant 10$	1
5	11	$p = 100$	
6	14	$r_1 \leqslant r_2 \leqslant \ldots \leqslant r_n$	
7	19	$k_{\max} = n$	
8	33	Нет дополнительных ограничений	1, 2, 3, 4, 5, 6, 7

Пример

стандартный ввод	стандартный вывод
1 5 2 3 2100 60 1980 2254 1870 2812 1760 WDWLW	2

Задача 6. Зеркальный лес

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

В древнем Зеркальном лесу разбросаны n волшебных камней. Странствующий чародей Элдан находится в точке S и хочет попасть на священную поляну в точке T. Идти пешком нельзя — лес защищён древним проклятием.

Зато камни откликаются на зов с любого расстояния: подходить к камню не нужно, Элдан касается его магией, не сходя с места. Если, стоя в точке Q, он так коснётся i-го камня, расположенного в точке P_i, то мгновенно телепортируется в точку Q', симметричную точке Q относительно P_i (то есть P_i — середина отрезка QQ'). После этого камень рассыпается в пыль, поэтому каждый камень можно использовать не более одного раза. Какие камни использовать и в каком порядке Элдан выбирает сам (использовать все камни необязательно).

Определите, может ли Элдан попасть из точки S ровно в точку T, и если может — укажите, каких камней и в каком порядке для этого коснуться. Делать это как можно меньшим числом касаний не требуется: подойдёт любой подходящий путь.

Формат входных данных

В первой строке задано одно целое число n — количество волшебных камней ($1 \leqslant n \leqslant 29$).

Во второй строке заданы два целых числа s_x, s_y — координаты точки S ($|s_x|, |s_y| \leqslant 10^7$).

В третьей строке заданы два целых числа t_x, t_y — координаты точки T ($|t_x|, |t_y| \leqslant 10^7$).

В каждой из следующих n строк заданы два целых числа x_i, y_i — координаты i-го камня P_i. Несколько камней могут находиться в одной точке ($|x_i|, |y_i| \leqslant 10^7$).

Формат выходных данных

Если Элдан может попасть из S в T, выведите в первой строке `YES`, во второй — число касаний k ($0 \leqslant k \leqslant n$), а в третьей — k различных целых чисел от 1 до n: номера камней в том порядке, в котором Элдан их касается.

Если попасть из S в T нельзя, выведите единственное слово `NO`.

Минимизировать число касаний не требуется: подойдёт любая допустимая последовательность касаний. При $k = 0$ третья строка должна быть пустой.

Система оценки

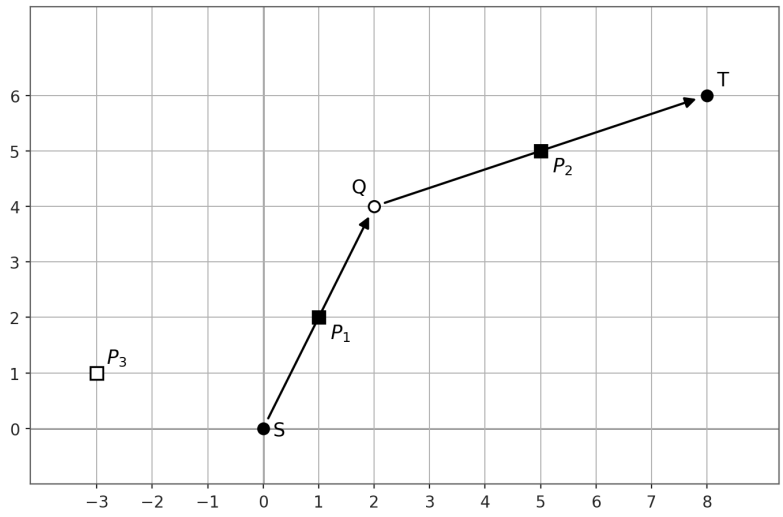
Группа	Баллы	Дополнительные ограничения		Необходимые подзадачи	Комментарий
		n	$\lvert x_i\rvert, \lvert y_i\rvert$		
1	0	–	–	–	Тесты из условия.
2	4	$n \leqslant 2$	–	1	
3	6	$n \leqslant 8$	–	1 – 2	
4	7	$n \leqslant 13$	–	1 – 3	
5	6	$n \leqslant 17$	–	1 – 4	
6	7	–	–	1 – 2	Камни расположены не более чем в 5 различных точках.
7	6	–	$\leqslant 10^3$	1 – 2	Все камни лежат на одной прямой.
8	7	–	$\leqslant 15$	1 – 2	$\lvert s_x\rvert, \lvert s_y\rvert, \lvert t_x\rvert, \lvert t_y\rvert \leqslant 500$
9	9	$n \leqslant 24$	–	1 – 5	
10	9	$n \leqslant 26$	–	1 – 5, 9	
11	10	$n \leqslant 27$	–	1 – 5, 9 – 10	Потестовая оценка: 1 балл за тест.
12	13	$n \leqslant 28$	–	1 – 5, 9 – 11	Потестовая оценка: 1 балл за тест.
13	16	–	–	1 – 12	Потестовая оценка: 1 балл за тест.

Примеры

стандартный ввод	стандартный вывод
3 0 0 8 6 1 2 5 5 -3 1	YES 2 1 2
3 1 1 5 3 4 5 2 1 3 4	YES 3 2 3 1
3 0 0 1 2 4 4 -1 3 2 -6	NO

Замечание

Пример 1



Пример 2

