

Задача А. Анизотропные простые числа

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 1024 мегабайта

Назовём *анизотропными* простые числа, сумма всех делителей которого не делится на количество этих делителей.

Задано целое число n , найти ближайшее к нему анизотропное простое число.

Формат входных данных

Входные данные содержат одно целое число n ($1 \leq n \leq 10^9$).

Формат выходных данных

Выведите одно целое число — ближайшее к заданному анизотропное простое число. Если таких чисел несколько, выведите наибольшее из них.

Пример

стандартный ввод	стандартный вывод
2	2

Задача В. Буду ли я участвовать?

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 1024 мегабайта

Согласно правил ICPC на сезон 2025-2026, участие в соревнованиях возможно в соответствии со следующими правилами.

1. При выступлении в прошлом в двух и более финалах участвовать в соревнованиях запрещено.
2. При выступлении в прошлом в шести и более региональных контекстах участвовать в соревнованиях сезона запрещено.
3. При выступлении в прошлом ровно в пяти региональных контекстах применяется правило 5.
4. Для родившихся ранее 2002 года и при этом впервые поступивших в университет ранее 2021 года, процедура допуска определяется правилом 5. В случае, если хотя бы одно из этих двух утверждений — про дату рождения или про дату поступления — неверно, участвовать в соревнованиях сезона разрешено.
5. Если закончено 9 или более семестров, участвовать в соревнованиях сезона запрещено. Если закончено 8 или менее семестров, то для решения вопроса о допуске к сезону требуется запрос руководству ICPC.

По информации с личной страницы на icpc.global: дате рождения, дате первого поступления в университет, количеству завершённых семестров, количеству участия в региональных контекстах и количеству участия в финалах определите, возможно ли участие в сезоне 2025-2026.

Формат входных данных

Первая строка входных данных содержит одно целое число — год рождения y ($1995 \leq y \leq 2011$). Вторая строка входных данных содержит одно целое число — год первого поступления в университет u ($2012 \leq u \leq 2025$). Третья строка входных данных содержит одно целое число s — количество завершённых семестров ($0 \leq s \leq 12$). Четвёртая строка входных данных содержит одно целое число f — количество участия в финалах ($0 \leq f \leq 3$). Пятая строка входных данных содержит одно целое число r — количество участия в региональных контекстах ($f \leq r \leq 7$).

Формат выходных данных

Если имеющиеся данные позволяют участвовать в соревнованиях сезона 2025-2026, выведите + (плюс).

Если имеющиеся данные запрещают участвовать в соревнованиях сезона 2025-2026, выведите - (минус).

Если вопрос о допуске будет решаться руководством ICPC, выведите ? (вопросительный знак).

Примеры

стандартный ввод	стандартный вывод
2002 2021 10 1 4	+
2001 2020 8 0 0	?
2003 2022 9 1 5	-

Задача С. Везёт слабейшим

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 1024 мегабайта

В Ночной Баскетбольной Ассоциации (НБА) распределение игроков между командами осуществляется с помощью драфта. Идея драфта заключается в том, что команды, хуже всего сыгравшие прошлый сезон, выбирают новых игроков первыми, тем самым имеют возможность получить более перспективных баскетболистов и за счёт этого в будущем построить сильный состав. Чтобы исключить явную «игру в поддавки», среди 14 команд, не вышедших в раунд игр на вылет, проводится лотерея за право первого выбора.

В урну для жеребьёвки помещается какое-то количество шаров с названиями команд внутри. Доля шаров с названиями команды с i -го снизу места равна a_i процентов, при этом последовательность a_i является невозрастающей, а её сумма равна 100. Значение чисел a_i и дальнейшая процедура выбора описаны в Коллективном Соглашении (договоре между командами и лигой, определяющем регламент НБА).

Вам поручили подготовить лотерею драфта. Вы прочитали раздел Коллективного Соглашения, определяющий значение чисел a_i , после чего хотите выбрать наименьшее количество шаров для того, чтобы можно было разложить названия команд в соответствии с требуемым соотношением.

Формат входных данных

Входные данные содержат 14 вещественных чисел a_i , расположенных в порядке невозрастания — процент шаров с названием команды, занявшей i -е снизу место ($1 \leq a_i \leq 100$, сумма всех a_i равна 100, все a_i заданы ровно с одним знаком после десятичной точки).

Формат выходных данных

Выведите одно целое число — наименьшее количество шаров для того, чтобы можно было нанести на них названия команд в соответствии с требуемым соотношением.

Пример

стандартный ввод	стандартный вывод
14.0 14.0 14.0 12.5 10.5 9.0 7.5 6.0 4.5 3.0 2.0 1.5 1.0 0.5	200

Задача D. Грузим файл

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	1024 мегабайта

В мессенджере «Диаграм» при отправке файла прогресс загрузки изображается следующей анимацией: процент загрузки отображается закрашенной дугой окружности радиуса 1 с центром в точке $O(0, 0)$, причём файл, загруженный на k процентов, соответствует k процентам полной окружности (то есть при полной загрузке файла закрашена вся окружность, при загрузке, например, на 80 процентов закрашена дуга в 288 градусов, при загрузке на 25 процентов — дуга в 90 градусов и так далее).

В этой задаче положение точки на окружности будем задавать её полярным углом — углом a в градусах между направлением из начала координат к этой точке и положительным направлением оси Ox ($0 \leq a < 360$).

Точка s , от которой откладывается дуга, в момент начала загрузки находится в точке с полярным углом 0, затем всё время загрузки движется против часовой стрелки с постоянной угловой скоростью v градусов в секунду.

Иначе говоря, рассмотрим произвольный момент времени t между началом и завершением загрузки включительно. В момент времени t начало дуги находится в текущем положении точки s , сама дуга имеет угловую величину, определённую текущим процентом загрузки, и откладывается от точки s против часовой стрелки. При этом закрашена только сама дуга (включая оба её конца), а остальные точки в момент времени t закрашенными не являются, даже если они были закрашены в предыдущие моменты времени.

Вам дано время d в секундах, за которое файл будет загружен и значение v . Определить, как будет выглядеть диаграмма в момент достижения процента загрузки p .

Формат входных данных

Первая строка входных данных содержит три целых числа d , v и p ($10 \leq d \leq 10^9$, $1 \leq v \leq 359$, $1 \leq p \leq 99$) — количество секунд, в течение которого файл загружается, угловая скорость точки s в анимации загрузки, в градусах в секунду, и процент загрузки, для которого требуется определить положение дуги.

Формат выходных данных

Выведите два числа — положение точки s , от которой откладывается дуга, и положение второго конца дуги, с абсолютной или относительной погрешностью не хуже 10^{-4} .

Пример

стандартный ввод	стандартный вывод
2025 10 12	270.000000 313.200000

Задача Е. Добиться делимости

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 1024 мегабайта

Вам дана десятичная запись целого положительного числа n , не содержащая ведущих нулей.

Требуется увеличить или уменьшить ровно одну цифру на 1 так, чтобы число делилось на 7 (и при этом по-прежнему не содержало ведущих нулей). Сколькими способами это можно сделать?

Формат входных данных

Входные данные содержат одно целое число n ($1 \leq n < 10^{10^5}$), записанное без ведущих нулей.

Формат выходных данных

Выведите одно целое число — количество способов увеличить или уменьшить ровно одну цифру на 1 так, чтобы число делилось на 7.

Примеры

стандартный ввод	стандартный вывод
17	0
1002	2

Задача F. Если заказчик требует...

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 1024 мегабайта

ООО «Рога и Копыта» организовала страйкбольный клуб и проводит в нем соревнования. Для простоты, будем считать, что страйкбольный клуб представляет собой поле $n \times m$, разделенное на квадраты 1×1 . Каждому квадрату приписано число, которое обозначает удобство этого квадрата.

Правила соревнований следующие: на территории клуба выделяется некоторый прямоугольник (каждая из его сторон должна быть не меньше 2), после чего две команды занимают две пары противоположенных углов этого прямоугольника и начинают биться друг с другом. Более формально, выбирается 4 числа, i_1, i_2, j_1, j_2 , $1 \leq i_1 < i_2 \leq n$, $1 \leq j_1 < j_2 \leq m$, после чего первая команда (назовем ее красной) занимает квадраты (i_1, j_1) и (i_2, j_2) , а вторая (синяя) — квадраты (i_1, j_2) и (i_2, j_1) .

В ближайшее воскресенье страйкбольный клуб арендовало ЗАО «12 стульев» для проведения корпоратива, причем за красную команду будет выступать сам гендиректор, Остап Ибрагимович. В соответствии с неформальными договорённостями при заключении договора аренды, председатель страйкбольного клуба Фунт очень хочет, чтобы красная команда победила. Помогите ему выделить прямоугольник таким образом, чтобы суммарное удобство стартовых квадратов красной команды, было больше, чем сумма удобств стартовых квадратов синей команды.

Формат входных данных

Первая строка входных данных содержит два числа, n и m , $n \geq 2, m \geq 2, 4 \leq nm \leq 10^6$. Следующие n строк содержат по m чисел каждая, где j -е число на i -й строке обозначает удобство квадрата с координатами (i, j) . Гарантируется, что все удобства квадратов не превосходят по модулю 10^{15} .

Формат выходных данных

Вывести четыре числа, i_1, i_2, j_1, j_2 , удовлетворяющие условию задачи. Если такую четверку найти невозможно, вывести -1.

Примеры

стандартный ввод	стандартный вывод
3 3 1 2 3 4 5 6 7 8 10	1 3 1 3
2 2 1 100 100 1	-1

Задача G. Ё-тауэр

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 1024 мегабайта

В очередном космическом квесте события происходят в очень высоком здании на поверхности далёкой планеты. В плане здание выглядит как русская буква «Ё», поэтому оно получило название «Ё-тауэр». Точки этой буквы — скоростные лифты, соединяющие этажи.

По сюжету игры персонаж должен расследовать загадочное исчезновение инженера, отвечавшего за функционирование электроники в здании. Расследование осложнял тот факт, что без этого инженера никто ничего толком не мог ответить про Ё-тауэр, включая количество этажей.

Перед исчезновением инженер поехал проверять что-то на последнем этаже здания. При внимательном осмотре одного из лифтов нашлась записка, состоящая из длинной строки десятичных цифр. Почерк этой записки был идентифицирован как почерк пропавшего инженера.

Персонал здания рассказал, что пропавший инженер при поездках в лифте записывал в строчку номера этажей, на которых останавливался лифт (чтобы потом сверить записи с логами регистрационных систем), однако из-за экономии бумаги не ставил пробелов между цифрами. Вы предположили, что записка была оставлена инженером либо при поездке с какого-то этажа на самый верхний, либо при поездке с самого верхнего этажа на один из нижних.

Зная, что пропавший был очень аккуратен и никогда не делал ошибок в записях, вы хотите по данной записке выяснить, какое наименьшее количество этажей может быть в здании Ё-тауэра.

Формат входных данных

Первая строка входных данных содержит непустую цифровую строку s , состоящую из цифр от 1 до 9. Длина строки не превосходит 10^5 .

Формат выходных данных

Выведите одно число — ответ к задаче.

Пример

стандартный ввод	стандартный вывод
142857	57

Задача Н. Желаете подешевле?

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 4 секунды
Ограничение по памяти: 1024 мегабайта

Требуется составить римское число из других римских чисел, стартуя с пустой строки. Разрешается приписывать корректное римское число к корректному римскому числу спереди или сзади, а также удалять у строки, являющейся корректным римским числом, префикс или суффикс, который является корректным римским числом, при условии, что результат этих операций будет корректным римским числом. Например, к **C** можно приписать **I** справа (получим римское 101), но нельзя приписать **I** слева (числа **IC** не существует). Из числа **MMXXV** можно получить **XXV** удалением префикса **MM**, или же **M** удалением суффикса **MXXV**.

По умолчанию цена добавления и удаления каждого римского числа равна его значению (то есть, например, удаление префикса **MM** стоит 2000, удаление суффикса **MXXV** стоит 1025). Однако для некоторых чисел цена добавления и удаления изменена (причём цена добавления и цена удаления может различаться). Цена добавления или цена удаления не зависит от того, с какой стороны число было дописано или удалено.

Вам задан список чисел (в десятичной записи) с изменёнными ценами добавления и удаления. Также вам заданы несколько чисел в десятичной записи. Для каждого из этих чисел найдите наименьшую цену, за которую можно построить это число, начиная с пустой строки.

Напоминаем современный способ записи римских чисел (таблица взята из Википедии):

Значение разряда	Тысячи	Сотни	Десятки	Единицы
1	M	C	X	I
2	MM	CC	XX	II
3	MMM	CCC	XXX	III
4		CD	XL	IV
5		D	L	V
6		DC	LX	VI
7		DCC	LXX	VII
8		DCCC	LXXX	VIII
9		CM	XC	IX

Заметим, что:

- Числа 4, 9, 40, 90, 400 и 900 записываются в реверсивной нотации, где первый символ вычитается из второго (например, для 40 (**XL**) '**X**' (10) вычитается из '**L**' (50)). Это **единственные** места, где реверсивная нотация используется.
- Число, содержащее несколько десятичных цифр, строится дописыванием римского эквивалента каждой цифры от старшего разряда к младшему.
- Если в десятичном разряде стоит 0, никаких цифр в этом разряде в римском представлении не пишется.
- Наибольшее число, которое может быть представлено в римской системе счисления — это число 3,999 (**MMMCMXCIX**).

Формат входных данных

Первая строка входных данных содержит одно целое число n ($0 \leq n \leq 3\,999$) — количество чисел, для которых заданы цены добавления и удаления. Каждая из последующих n строк содержит по три целых числа — число x ($1 \leq x \leq 3\,999$), для римской записи которого определяются новые цены добавления и удаления, и два числа add_x и del_x — новые цены для приписывания и удаления римской записи x , соответственно ($1 \leq add_x, del_x \leq 4\,000$). Для тех x , которые не указаны в этих n строках, $add_x = del_x = x$. Далее идёт строка, содержащая одно целое число q ($1 \leq q \leq 3\,999$) — количество запросов. Каждая из последующих q строк задаёт один запрос и содержит одно целое число y_i ($1 \leq y_i \leq 3\,999$).

Формат выходных данных

Для каждого запроса выведите одно целое число — минимальную цену, за которую римскую запись числа из этого запроса можно построить, начиная с пустой строки.

Пример

стандартный ввод	стандартный вывод
2	402
100 1 2	18
1200 2 3	
2	
500	
3000	

Задача I. Загадано основание

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	1024 мегабайта

Это интерактивная задача

Загадано основание позиционной системы счисления b ($2 \leq b \leq 65\,536$). Вы можете задавать запросы, содержащие одно целое число x ($1 \leq x \leq 65\,536$) и получить ответ — остаток от деления на 2 количества цифр в b -ичной записи числа x .

Например, если $b = 10$ и $x = 2025$, то ответ будет 0 (так как в 2025 чётное количество цифр). А если $b = 16$ и $x = 2025$, то ответ будет 1 (7E9, нечётное количество цифр). А если $b = 65\,536$ и $x = 2025$, то ответ будет 1 (число 2025 является однозначным в системе счисления с основанием 65 536, 1 — нечётное число).

Ваша задача — угадать b не более, чем за 19 запросов.

Протокол взаимодействия

Взаимодействие начинает программа жюри, выводя одно целое число t ($1 \leq t \leq 1600$) — количество сценариев. Далее начинается обработка сценариев.

В каждом сценарии взаимодействие начинает ваша программа, задавая запрос в формате $? x$, ($1 \leq x \leq 65\,536$). Ответом на запрос будет 0, если запись числа x в b -ичной системе без ведущих нулей имеет чётное количество цифр, и 1 в противном случае.

Как только ваша программа готова вывести ответ, она должна вывести его в формате $! b$, где b — предполагаемое значение основания системы счисления. Если при обработке сценария не произошло ошибок, и если этот сценарий не последний, то после этого сразу же начинается следующий сценарий. Вывод ответа запросом не считается.

Гарантируется, что значение b не меняется в процессе взаимодействия (то есть что интерактор не адаптивный).

Не забывайте выводить после вывода запроса и ответа перевод строки и сбрасывать буфер вывода вызовом функции `flush` используемого вами языка программирования.

Пример

стандартный ввод	стандартный вывод
1	$? 2025$
0	$? 10$
0	$? 12$
0	$? 700$
1	$! 10$

Замечание

В примере взаимодействия программа участника вывела ответ наугад (например, подошло бы ещё основание 9). Участнику просто повезло.

Задача J. Инфляция и трамвай

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 1024 мегабайта

Трамвайная сеть столицы Байтландии состоит из n остановок и m двусторонних трамвайных перегонов, соединяющих остановки. При этом между любыми двумя остановками можно проехать по сети перегонов.

Городской департамент транспорта изучает влияние курса байтландского тугрика к российскому рублю на стоимость обслуживания сети. При курсе байтландского тугрика, равном x , стоимость обслуживания i -го перегона равна $a_i + b_i \cdot x$ (коэффициент b_i может быть и отрицательным; если стоимость отрицательная, то эксплуатация перегона приносит доход), где a_i и b_i — параметры, зависящие от перегона.

Новый глава Департамента Транспорта, автомобилист с многолетним стажем, планирует закрыть все «лишние» перегоны, оставив ровно $n - 1$ перегон так, чтобы между любыми двумя остановками всё ещё можно было проехать на трамвае. Для фиксированного x обозначим значение $f(x)$ как минимальную суммарную стоимость обслуживания перегонов после реализации плана департамента транспорта.

Так как курс байтландского тугрика нестабилен, вам задаётся диапазон курса от ℓ до r включительно. Ваша задача — вычислить значение $\min(f(\ell), f(\ell + 1), \dots, f(r))$.

Формат входных данных

Входные данные состоят из нескольких тестовых примеров.

Первая строка каждого тестового примера содержит четыре целых числа n , m , ℓ и r , указывающих количество остановок, количество перегонов и диапазон курса, соответственно ($2 \leq n \leq 10^5$, $n - 1 \leq m \leq 2 \times 10^5$, $0 \leq \ell \leq r \leq 10^6$).

Каждая из последующих m строк описывает один перегон и содержит четыре целых числа u_i , v_i , a_i и b_i , указывающих, что i -й перегон соединяет остановки u_i и v_i и имеет параметры a_i и b_i ($1 \leq u_i, v_i \leq n$, $u_i \neq v_i$, $1 \leq a_i \leq 10^6$, $-10^6 \leq b_i \leq 10^6$). Гарантируется, что между любыми двумя остановками можно проехать по сети перегонов. Также гарантируется, что сумма n по всем тестовым примерам не превышает 10^6 , а сумма m не превышает 2×10^6 .

Входные данные заканчиваются тестовым примером с $n = m = \ell = r = 0$, обрабатывать который не требуется.

Формат выходных данных

Для каждого тестового примера выведите одно целое число — требуемый минимум.

Пример

стандартный ввод	стандартный вывод
5 6 2 5 1 2 4 1 2 3 6 -1 3 4 2 2 4 5 2 -1 5 1 6 3 2 4 4 -1 5 7 1 5 1 2 1 1 2 3 2 2 3 4 1 -10 3 4 2 10 5 1 1 10 2 4 2 -10 1 2 1 0 0 0 0 0	6 -45

Задача К. Йошкар-Ола — Москва

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	5 секунд
Ограничение по памяти:	1024 мегабайта

На станции Йошкар-Ола тестируется инновационная полностью автоматизированная система загрузки банок с мёдом в железнодорожные вагоны для последующей транспортировки.

Под погрузкой находится состав из n вагонов. Автоматизированная система погрузки выполняет следующие операции:

1. Погрузить по w банок, каждая из которых имеет ценность v , в вагоны с l -го по r -й включительно.
2. Определить ценность k -й самой дешёвой банки, находящейся в m -м вагоне. Гарантируется, что при выполнении этой операции в вагон погружено не менее k банок.

Ваша задача — промоделировать эти операции.

Формат входных данных

Первая строка входных данных содержит два целых числа — количество вагонов в составе n ($1 \leq n \leq 2 \cdot 10^5$) и количество операций q ($2 \leq q \leq 2 \cdot 10^5$).

Каждая из последующих q строк содержит описание операций, перечисленных в порядке выполнения. i -я из этих строк задаёт i -ю операцию и начинается целым числом t_i ($1 \leq t_i \leq 2$), задающего тип операции.

Если $t_i = 1$, то происходит погрузка, и далее в строке содержатся четыре целых числа l_i , r_i , v_i и w_i ($1 \leq l_i \leq r_i \leq n$, $1 \leq v_i \leq 10^9$, $1 \leq w_i \leq 10^4$) — левая и правая граница диапазона вагонов, ценность одной банки мёда и количество загружаемых банок, соответственно.

Если $t_i = 2$, то происходит проверка ценности, и далее в строке содержатся два целых числа m_i ($1 \leq m_i \leq n$) и k_i ($1 \leq k_i \leq s$, где s_{m_i} — текущее количество банок в вагоне m_i).

Гарантируется, что во входных данных присутствует как минимум одна команда типа 2.

Формат выходных данных

Для каждой команды типа 2 выведите требуемую ценность в отдельной строке.

Пример

стандартный ввод	стандартный вывод
6 8	1
1 1 4 1 2	3
1 2 6 3 4	2
2 4 2	3
2 4 3	1
1 1 5 2 1	
2 4 3	
2 6 3	
2 1 1	

Задача L. Кузнечик и отмеченные точки

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2.5 секунд
Ограничение по памяти: 1024 мегабайта

Рассмотрим координатную прямую, на которой кузнечик стартует из точки 0 и должен достичь точки ℓ .

За один ход кузнечик может переместиться из текущей позиции x в позицию $x + 1$ или $x + 2$.

На координатной прямой отмечены n точек x_1, x_2, \dots, x_n таких, что $1 \leq x_i \leq \ell - 1$.

Назовём *маршрутом* последовательность допустимых прыжков, ведущую из точки 0 в точку ℓ .

Для каждого маршрута p определим $c(p)$ как количество отмеченных точек, которые были посещены в процессе следования по маршруту (т.е. координаты которых встречаются в последовательности позиций маршрута).

Для данного k найдите сумму:

$$\sum_{p \in \mathcal{P}} (c(p))^k,$$

где \mathcal{P} — множество всех маршрутов из 0 в ℓ .

Так как ответ может быть очень большим, выведите результат по модулю 998 244 353.

Формат входных данных

В первой строке записано одно целое число t ($1 \leq t \leq 10\,000$) — количество тестовых примеров. Далее следуют тестовые примеры.

Первая строка каждого тестового примера содержит три целых числа ℓ, n, k ($2 \leq \ell \leq 10^9$, $1 \leq n \leq 100\,000$, $1 \leq k \leq 20$) — координата конечной точки, количество отмеченных точек и степень соответственно.

Вторая строка содержит n различных целых чисел x_1, x_2, \dots, x_n ($1 \leq x_i \leq \ell - 1$) — координаты отмеченных точек, в порядке возрастания.

Гарантируется, что сумма n по всем тестовым примерам не превосходит 100 000

Формат выходных данных

Для каждого набора входных данных выведите одно целое число — остаток от деления требуемой суммы на 998 244 353.

Пример

стандартный ввод	стандартный вывод
3	64
10 1 1	64
5	1048578
5 4 2	
1 2 3 4	
3 2 20	
1 2	

Задача М. Лист клетчатой бумаги

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 1024 мегабайта

Назовём *горизонтальной гистограммой* на клетчатой бумаге фигуру высоты m клеток, в которой i -я сверху строка имеет $a_i \geq 0$ столбцов. Если существует способ представить горизонтальную гистограмму как объединение конечного числа неперекрывающихся прямоугольников 1×2 и 2×1 , то назовём последовательность a_1, \dots, a_m *правильной*.

Вам задана целочисленная последовательность b_1, \dots, b_n . Найдите количество таких пар (ℓ, r) , что $1 \leq \ell \leq r \leq n$ и непрерывная подпоследовательность b_ℓ, \dots, b_r из $r - \ell + 1$ элементов будет правильной.

Формат входных данных

Первая строка входных данных содержит одно целое число t — количество тестовых примеров ($1 \leq t \leq 100$).

Далее следуют тестовые примеры. Каждый тестовый пример состоит из двух строк. Первая строка содержит целое положительное число n — длину последовательности b ($1 \leq n \leq 5 \cdot 10^5$).

Вторая строка содержит n целых чисел b_i ($0 \leq b_i \leq 10^9$).

Гарантируется, что сумма значений n по всем тестовым примерам не превосходит 10^6 .

Формат выходных данных

Для каждого тестового примера выведите одну строку — ответ на вопрос задачи для этого тестового примера.

Пример

стандартный ввод	стандартный вывод
3	7
5	28
7 8 8 7 5	1
9	
5 9 2 10 6 6 0 8 11	
1	
0	