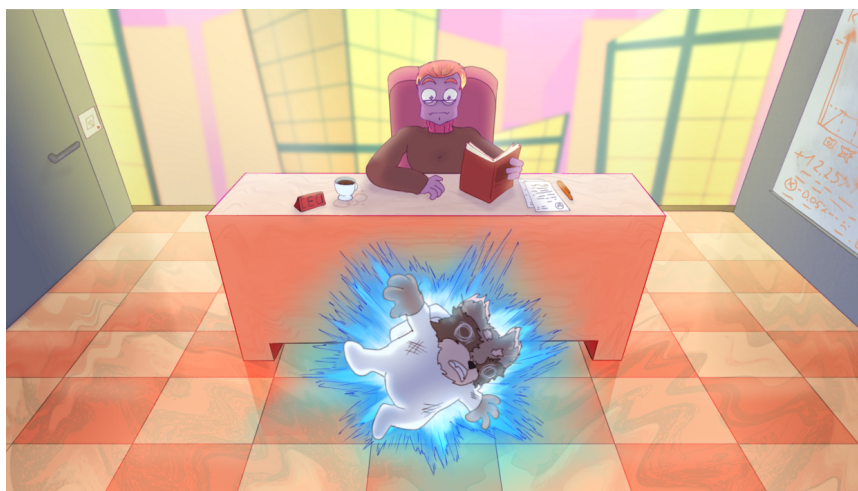


Задача А. Неожиданные последствия укуса

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

После героического укуса Енот отлетел настолько далеко в прошлое, насколько это было возможно. Перегрев машины времени послужил причиной Большого взрыва, который, в свою очередь, откинул мохнатого обратно в будущее, но теперь — ещё более далёкое. Отойдя от временной перегрузки, Енот понял, что находится в кабинете подросткового Лжепетя, главы корпорации Turbopallascats.



Лжепетя был невероятно рад видеть Енота живым. Зверек был насторожен, однако директор продемонстрировал ему, как изменился он сам и весь мир за последнее время. Turbopallascats теперь занимается собственными разработками и помощью людям, пострадавшим от Гамена — прошлого директора компании. Лжепетя хотел показать всё это Еноту вживую, однако, кажется, он потерял ключ-карту, необходимую для выхода из офиса. Ну что же, придётся воспользоваться аварийным выходом — воротами с функцией проверки интеллекта.

На этот раз на воротах был изображён массив a , состоящий из n целых чисел. За одно действие можно выбрать любые два числа a_i и a_j такие, что $i \neq j$, и заменить число a_i на сумму a_i и a_j . Ворота откроются только тогда, когда все числа в массиве будут различны. Программный модуль, который находится внутри ворот, не рассчитан на большие числа, поэтому в любой момент времени для всех чисел должно выполняться $|a_i| \leq 10^{18}$.

Енот уже хотел по старой памяти покусать ворота, однако Лжепетя остановил его — ворота были под напряжением. Чтобы не испытывать терпение Енота, он решил открыть ворота не более чем за m действий, и у него это получилось. А получится ли у вас? Лжепетя не жульничал, а значит гарантируется, что сделать все числа в массиве различными за заданное количество действий возможно.

Формат входных данных

В первой строке находится одно целое число t ($1 \leq t \leq 3 \cdot 10^5$) — количество тестовых примеров. Далее следуют t тестовых примеров, заданных описанным ниже способом.

В первой строке тестового примера находятся два целых числа n и m ($1 \leq n \leq 3 \cdot 10^5$, $n + c - 1 \leq m \leq 6 \cdot 10^5$, где c — количество нулей в массиве a) — количество элементов в массиве a и количество действий.

Во второй строке тестового примера находятся n чисел a_i ($-10^9 \leq a_i \leq 10^9$).

Обозначим сумму n по всем тестовым примерам как S_n , а сумму m по всем тестовым примерам как S_m . Тогда гарантируется, что $S_n \leq 3 \cdot 10^5$, $S_m \leq 6 \cdot 10^5$.

Формат выходных данных

Для каждого из тестовых примеров выведите ответ в следующем формате.

Для i -го тестового примера в первой строке выведите целое число s ($0 \leq s \leq m$) — количество совершённых действий. Затем, в каждой из следующих k строк выведите два целых числа i и j — описание k -го действия.

Система оценки

Пусть c — количество нулей в текущем тестовом примере.

№	Дополнительные ограничения	Баллы за подзадачу	Необходимые подзадачи
1	$S_n \leq 2, m = 10$	11	
2	$a_i \geq 0, n \leq 20, m = 6 \cdot 10^5$	14	
3	$a_i > 0$	15	
4	$a_i \geq 0$	20	2-3
5	Все числа в массиве равны	10	
6	Все числа в массиве равны нулю, кроме одного	11	
7	Без дополнительных ограничений	19	1-6

Пример

стандартный ввод	стандартный вывод
4	3
6 40	1 2
3 3 4 6 5 3	1 3
4 20	6 3
0 2 0 -1	2
9 100	3 2
2 -2 5 0 -1 0 0 2 -2	2 4
3 10	7
1 2 3	2 3
	4 1
	5 4
	6 5
	6 9
	4 1
	1 3
	0

Замечание

Рассмотрим первый тестовый пример.

- После первого действия к первому числу массива было прибавлено второе: $3 + 3 = 6$. Итоговый массив: $[6, 3, 4, 6, 5, 3]$.
- После второго действия к первому числу массива было прибавлено третье: $6 + 4 = 10$. Итоговый массив: $[10, 3, 4, 6, 5, 3]$.
- После третьего действия к шестому числу массива было прибавлено третье: $3 + 4 = 7$. Итоговый массив: $[10, 3, 4, 6, 5, 7]$.

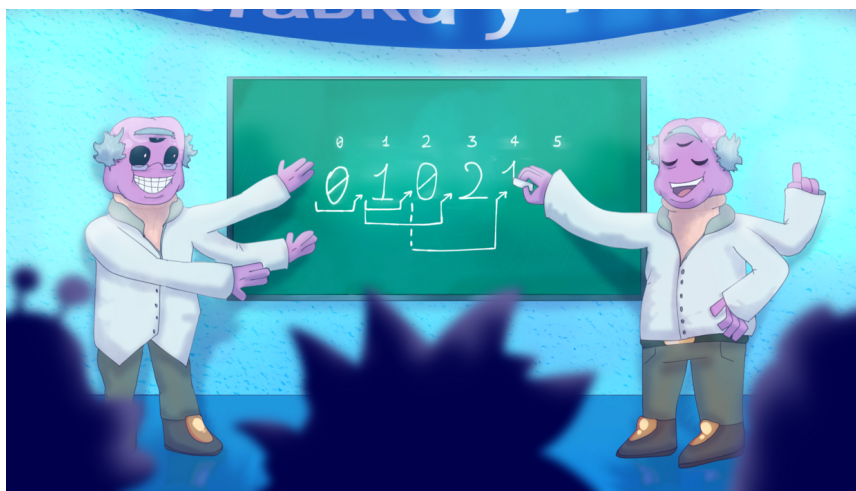
Спустя три действия все числа в массиве a стали различными.

В четвёртом тестовом примере все числа в массиве уже различные, поэтому применять действия не нужно.

Задача В. Очередной прорыв в науке

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

Марсианские профессора — настоящие мастодонты своего дела. Они завоевали не один десяток наград во всевозможных сферах — от антропологии, биологии и социологии до физики и информатики. И сегодня марсианские ученые представляют научному комитету свою новую монументальную работу в области математики. Они изобрели концептуально новый вид последовательностей — *марсианский ряд*!



Марсианский ряд обозначается буквой M . Первое число марсианского ряда равно $M_0 = 0$. Число M_i определяется как минимальное целое неотрицательное число, которое не встречается в марсианском ряду на отрезке от $\lfloor \frac{i}{2} \rfloor$ до $i - 1$ включительно. Более формально,

$$M_i = \text{MEX} (M_j \text{ по всем } \lfloor \frac{i}{2} \rfloor \leq j \leq i - 1),$$

где MEX — минимальное целое неотрицательное число, не представленное в данном множестве.

Комиссия с интересом выслушала доклад марсианских профессоров, после которого у неё появилось целых q вопросов. По забавному совпадению, все вопросы имели одинаковый формат:

«Подскажите, а если мы для некоторых l и r возьмём числа $M_l, M_{l+1}, \dots, M_r$ и отсортируем их в порядке неубывания, какое число будет находиться на k -й позиции слева?»

Вопросы весьма интересны, однако из-за короткого периода концентрации внимания в будущем марсианским профессорам отводится всего одна минута на весь доклад и ответы на вопросы, и они боятся не успеть произвести все вычисления вовремя. Сможете ли вы им помочь?

Формат входных данных

В первой строке находится одно целое число q ($1 \leq q \leq 10^5$) — количество вопросов от комиссии. Далее следуют параметры вопросов.

В каждой из следующих q строк находятся три целых числа l , r и k ($0 \leq l \leq r \leq 10^9$, $1 \leq k \leq r - l + 1$) — параметры вопроса комиссии.

Формат выходных данных

Для каждого вопроса выведите одно целое число в отдельной строке — ответ на очередной вопрос комиссии.

Система оценки

№	Дополнительные ограничения	Баллы за подзадачу	Необходимые подзадачи
1	$q, l, r \leq 200$	5	
2	$q, l, r \leq 2000$	13	1
3	$l, r \leq 2 \cdot 10^5$	17	1-2
4	Сумма по всем $r - l + 1 \leq 10^6$	20	1
5	$k = r - l + 1$	5	
6	$k = 1$	17	
7	$k \leq 5$	11	6
8	Нет дополнительных ограничений	12	1-7

Пример

стандартный ввод	стандартный вывод
4	0
0 1 1	1
1 4 2	3
0 5 6	1
0 4 4	

Замечание

Первые шесть значений марсианского ряда принимают следующие значения:

$$M_0 = 0$$

$$M_1 = \text{MEX}(M_0) = \text{MEX}(0) = 1$$

$$M_2 = \text{MEX}(M_1) = \text{MEX}(1) = 0$$

$$M_3 = \text{MEX}(M_1, M_2) = \text{MEX}(1, 0) = 2$$

$$M_4 = \text{MEX}(M_2, M_3) = \text{MEX}(0, 2) = 1$$

$$M_5 = \text{MEX}(M_2, M_3, M_4) = \text{MEX}(0, 2, 1) = 3$$

Для второго вопроса из тестового примера последовательность элементов равна 1, 0, 2, 1. Если отсортировать в порядке неубывания, то получим 0, 1, 1, 2. Число на второй позиции равно 1 — ответ на соответствующий вопрос.

Для третьего вопроса из тестового примера последовательность элементов равна 0, 1, 0, 2, 1, 3. Если отсортировать в порядке неубывания, то получим 0, 0, 1, 1, 2, 3. Число на шестой позиции равно 3 — ответ на соответствующий вопрос.

Задача С. Старые проблемы игрушечной логистики

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

Известный частный детектив Petya++, вместе со своим механическим сыном, роботом SAN, сегодня встречает в своём бюро гостей. К нему пожаловал старый друг — Петя, со своей женой Евой и дочкой Сашей. Пока взрослые обсуждали свои скучные взрослые темы и предавались воспоминаниям, дети занимались чем-то гораздо более интересным — разбирались с новой игрушечной железной дорогой!



Игрушечная железная дорога состоит из n станций, расположенных по кругу, где каждая станция пронумерована целыми числами от 1 до n по часовой стрелке, а также пластикового паровоза с одним вагончиком. Также в наборе с железной дорогой идут несколько грузовых кубиков одного из двух типов — лёгкие и тяжёлые.

В начале игры девочка Саша раскладывает грузы по начальным станциям. Каждый из этих грузов нужно отвезти в одну из соседних станций — либо по часовой стрелке, либо против. Для станции с номером i соседней по часовой стрелке будет станция с номером $i + 1$, а для станции с номером n соседней по часовой стрелке будет станция с номером 1. За один проезд между станциями можно перевезти от одного до двух лёгких грузов, либо один тяжёлый.

После расстановки робот SAN принимается продумывать стратегию, как быстрее всего развезти все грузы. Для того чтобы думать было проще, он вводит несколько правил:

1. Ездить без грузов скучно, поэтому поезд никогда не будет ехать пустым.
2. Поезд должен начать со станции номер 1 и в ней же закончить.
3. Каждый груз можно перевозить только один раз.
4. Если поезд должен перевезти груз из станции с номером a в станцию с номером b , то за время его перевозки он не может посетить никакие другие станции, кроме a и b .

Девочка Саша находит такие правила скорее усложняющими игру, однако делающими её чуть более интересной. Вместе они задались вопросом — какое минимальное количество поездок между соседними станциями понадобится, чтобы доставить все грузы в пункты назначения, не нарушая правил, и возможно ли это вообще?

Формат входных данных

Первая строка содержит одно целое число n ($3 \leq n \leq 4 \cdot 10^5$) — количество станций.

Каждая из последующих n строк содержит описание грузов, которые нужно доставить из станции с номером i ($0 \leq a_i^+, b_i^+, a_i^-, b_i^- \leq 10^9$), где:

- a_i^+ — число тяжёлых грузов, которые нужно доставить в следующую по часовой стрелке станцию.
- b_i^+ — число лёгких грузов, которые нужно доставить в следующую по часовой стрелке станцию.
- a_i^- — число тяжёлых грузов, которые нужно доставить в предыдущую по часовой стрелке станцию.
- b_i^- — число лёгких грузов, которые нужно доставить в предыдущую по часовой стрелке станцию.

Формат выходных данных

Если возможно доставить все грузы по назначению соблюдая все вышеописанные правила, выведите одно целое число — минимальное число перемещений паровоза. Если же это невозможно, выведите -1 .

Система оценки

№	Дополнительные ограничения	Баллы за подзадачу	Необходимые подзадачи
1	$n \leq 10^3; a_i^-, a_i^+ \leq 1; b_i^-, b_i^+ = 0$	10	
2	$\sum_{i=1}^n (a_i^- + a_i^+) \leq 10^5; b_i^-, b_i^+ = 0$	8	1
3	$a_i^-, a_i^+ \geq 1; b_i^-, b_i^+ = 0$	10	
4	$b_i^-, b_i^+ = 0$	9	1-3
5	$n \leq 5000; a_i^-, a_i^+ \leq 10^4; \sum_{i=1}^n (b_i^- + b_i^+) \leq 25$	12	1
6	$n \leq 5000; a_i^-, a_i^+, b_i^-, b_i^+ \leq 10^4$	20	1,5
7	$n = 3$	11	
8	Нет дополнительных ограничений	21	1-7

Примеры

стандартный ввод	стандартный вывод
5 0 2 1 0 1 0 0 3 1 4 0 2 0 2 3 0 0 1 0 4	18
3 2 0 1 0 1 0 2 0 2 0 1 0	-1
4 3 8 1 7 4 9 5 3 3 9 5 9 6 5 3 5	68

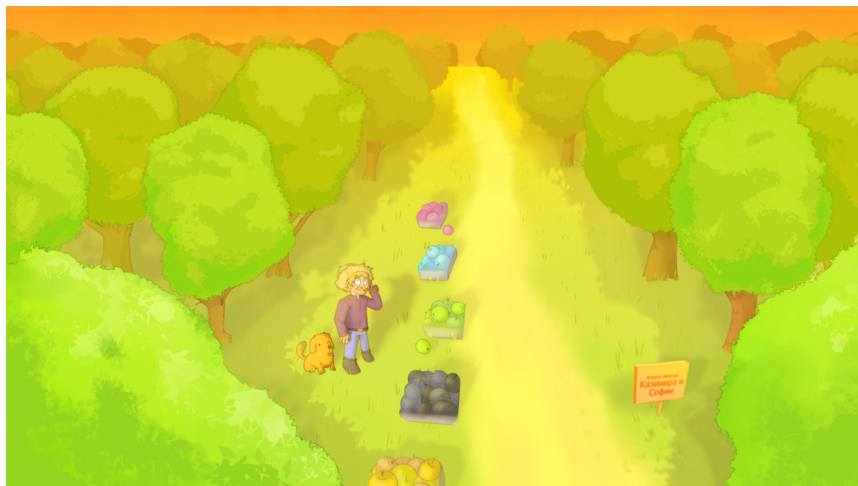
Замечание

В первом тестовом примере один из кратчайших маршрутов выглядит так: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 2 \rightarrow 1 \rightarrow 5 \rightarrow 1$.

Задача D. Садовник с Марса

Имя входного файла:	стандартный ввод или inputX.txt
Имя выходного файла:	стандартный вывод или outputX.txt
Ограничение по времени:	4 секунды
Ограничение по памяти:	256 мегабайт

Афанасий Казимирович — марсианский садовник, продолжающий дело своего отца, собрал урожай цветных марсианских яблок в этом году. Однако каждый знает, что мало вырастить хорошие яблоки — важно также уметь их продать. А покупатели будущего ценят эстетику не меньше вкуса!



Афанасий собрал n ящиков яблок. Каждый ящик содержит яблоки только одного определённого цвета, который можно описать целым положительным числом a_i . Садовник знает, что скоро приедет крупный покупатель, который выберет два различных целых числа l и r ($1 \leq l < r \leq n$) и купит все ящики с номерами от l до r включительно. Этот покупатель — большой ценитель искусства, поэтому Афанасий проводит уже не первый день, изучая, как бы угодить важному клиенту.

Марсианская теория цвета утверждает, что *сочетаемость* двух цветов A и B равна $A \oplus B$, где $\oplus$ означает операцию исключающего ИЛИ. Хотя эта теория и была раскритикована многими современниками, ожидаемый клиент относится к ней с большим уважением, поэтому Афанасий решил взять её за основу.

Исключающее ИЛИ — операция, которая на вход принимает два бита и возвращает бит 0, если биты на входе равны и 1 в противном случае. Например: $1 \oplus 0 = 1$, $1 \oplus 1 = 0$. Для применения побитового исключающего ИЛИ двух чисел эти числа сначала переводят в двоичную систему счисления, а затем применяют исключающее ИЛИ к каждому из разрядов. Например, $6 \oplus 3 = 5$, поскольку $6 = 110_2$, $3 = 11_2$. Применив исключающее ИЛИ поразрядно, получаем $5 = 101_2$:

$$\begin{array}{r} 110_2 \\ \oplus \quad 11_2 \\ \hline 101_2 \end{array}$$

В языке программирования Pascal побитовое исключающее ИЛИ чисел a и b обозначается $a \text{ xor } b$, а в языках Python и C++ — $a \wedge b$.

Марсианский садовник считает, что покупатель скорее пессимист, поэтому *красотой* купленного набора он посчитает минимальную сочетаемость цветов яблок из двух различных ящиков, купленных им, то есть величину $\min(a_i \oplus a_j)$ по всем $l \leq i < j \leq r$. Афанасий Казимирович ещё не знает, какие именно значения l и r выберет его клиент, поэтому он попросил вас посчитать суммарное значение красот для всех подходящих пар (l, r) .

Кроме того, Афанасию интересно, какое максимальное суммарное значение красот он мог бы получить, если бы он мог самостоятельно выбрать цвет яблок в каждом из n ящиков, если каждый цвет является целым положительным числом, не превосходящим $A_{\max}$.

Формат входных данных

Эта задача состоит из двух частей, каждая из которых имеет свой формат входных и выходных данных. Часть 1 является обычной задачей, а часть 2 — задачей с открытыми тестами. Ниже следует формат входных данных для каждой из частей.

Часть 1

В первой строке входных данных находится целое число q ($1 \leq q \leq 1000$) — количество тестовых примеров.

Далее следуют q тестовых примеров, заданных описанным ниже способом.

В первой строке тестового примера находится целое число n ($1 \leq n \leq 2 \cdot 10^5$) — количество ящиков яблок.

Во второй строке тестового примера находятся n чисел a_i ($1 \leq a_i \leq 10^9$) — цвет яблок в ящике номер i .

Обозначим сумму n по всем тестовым примерам как S_n . Тогда гарантируется, что $S_n \leq 2 \cdot 10^5$.

Часть 2

Входные данные находятся в файлах `input1.txt`, `input2.txt`, ..., `input5.txt`.

Первая строка входных данных содержит одно целое число t — номер теста (для примера из условия $t = 0$).

Вторая строка входных данных содержит три целых числа n , A_{max} и k — необходимое количество ящиков, количество возможных цветов яблок в ящиках и особое число для оценивания.

Дальнейшая часть входных данных описывает систему оценки. Значение этих чисел будет объяснено ниже.

Третья строка входных данных содержит $k + 1$ целое число p_i . Гарантируется, что $0 = p_1 < p_2 < \dots < p_k < p_{k+1}$. Гарантируется также, что сумма p_{k+1} по всем тестам равна 30.

Четвертая строка входных данных содержит $k + 1$ вещественное число s_i . Гарантируется, что $0 = s_1 \leq s_2 \leq \dots \leq s_k < s_{k+1}$.

Формат выходных данных

Часть 1

Для каждого из тестовых примеров выведите ответ в отдельной строке. Каждая строка должна содержать одно целое число — суммарное значение красот для всех подходящих пар (l, r) .

Часть 2

На проверку необходимо сдать выходные файлы с названиями `output1.txt`, `output2.txt`, ..., `output5.txt`, где выходной файл `outputX.txt` должен соответствовать входному файлу `inputX.txt`.

Выведите n целых чисел a_i — цвета яблок в ящиках.

Система оценки

Максимальный балл за часть 1 составляет 70, а максимальный балл за часть 2 составляет 30.

Часть 1

№	Дополнительные ограничения	Баллы за подзадачу	Необходимые подзадачи
1	$n \leq 100, S_n \leq 100$	6	
2	$n \leq 500, S_n \leq 500$	8	1
3	$n \leq 2 \cdot 10^3, S_n \leq 2 \cdot 10^3$	10	2
4	$a_i \leq 63$	8	
5	Все числа отсортированы	9	
6	Нет дополнительных ограничений	29	1-6

Часть 2

Если выходной файл не соответствует указанному формату выходных данных, то Вы получите 0 баллов за тест.

Если Вы вывели некорректный массив, то Вы также получаете 0 баллов за тест.

Пусть ans — суммарное значение красот в Вашем массиве. Если $ans \geq p_{k+1}$, то Вы получите s_{k+1} баллов за тест. Иначе, пусть pos — наименьшее целое положительное число такое, что $p_{pos} > ans$. Тогда Вы получите $s_{pos-1} + (s_{pos} - s_{pos-1}) \frac{ans - p_{pos-1}}{p_{pos} - p_{pos-1}}$ баллов за тест.

Баллы за каждый тест округляются вверх до сотых и суммируются. Правила округления таковы, что, например, при округлении числа 10.112 вверх до сотых получаем число 10.12.

Примеры

стандартный ввод или inputX.txt	стандартный вывод или outputX.txt
3 4 1 2 3 2 4 10 20 30 40 3 419103616 217819251 901224783	6 124 1632471330
0 5 10 2 0 18 30 0 2.5 6	1 7 3 2 6

Замечание

Первый тестовый пример относится к части 1, а второй тестовый пример относится к части 2.

В первом тестовом примере посчитаем красоту всех возможных наборов: $1 \oplus 2 = 3$, $\min(1 \oplus 2, 2 \oplus 3, 1 \oplus 3) = 1$, $\min(1 \oplus 2, 1 \oplus 3, 1 \oplus 2, 2 \oplus 3, 2 \oplus 2, 3 \oplus 2) = 0$, $\min(2 \oplus 3) = 1$, $\min(2 \oplus 3, 2 \oplus 2, 3 \oplus 2) = 0$, $\min(3 \oplus 2) = 1$. Суммарная красота равна $3 + 1 + 0 + 1 + 0 + 1 = 6$.

Во втором тестовом примере суммарное значение красот в массиве составляет 22. Рассмотрим, сколько баллов получит такое решение.

Поскольку $p_1 = 0$, $p_2 = 18$, $p_3 = 30$, то $pos = 3$. По формуле получим

$$s_2 + (s_3 - s_2) \frac{ans - p_2}{p_3 - p_2} = 2.5 + 3.5 \cdot \frac{4}{12} \approx 3.667.$$

Таким образом, итоговый балл равен 3.67.