

[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

А. Чубакка и число

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 256 мегабайт

Люк Скайуокер дал Чубакке целое число x . Чубакка не особо разбирается в числах, но обожает инвертировать в них цифры. Инвертировать цифру t — значит заменить её на цифру $9 - t$.

Помогите Чубакке из исходного положительного числа x получить минимальное **положительное** число, инвертировав некоторое (возможно, нулевое) количество цифр. Запись итогового числа не должна начинаться с нуля.

Входные данные

В первой строке содержится единственное целое число x ($1 \leq x \leq 10^{18}$) — число, которое Люк Скайуокер дал Чубакке.

Выходные данные

Выведите ответ на задачу — минимально возможное положительное число, которое может получить Чубакка после инвертирования некоторых цифр. Число не должно содержать ведущих нулей.

Примеры

входные данные	Скопировать
27	
выходные данные	Скопировать
22	
входные данные	Скопировать
4545	
выходные данные	Скопировать
4444	



[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

С. Троичный XOR

ограничение по времени на тест: 1 секунда

ограничение по памяти на тест: 256 мегабайт

Число называется *троичным*, если оно содержит только цифры 0, 1 и 2. Например, следующие числа являются троичными: 1022, 11, 21, 2002.

Вам задано длинное троичное число x . Первая (самая левая) цифра числа x гарантированно является 2, остальные цифры числа x могут быть 0, 1 или 2.

Определим троичную операцию XOR \odot над двумя троичными числами a и b (оба имеют длину n) как число $c = a \odot b$ длины n , где $c_i = (a_i + b_i) \% 3$ (где $\%$ — операция взятия по модулю). Другими словами, сложим соответствующие цифры и возьмем остатки от сумм при делении на 3. Например, $10222 \odot 11021 = 21210$.

Ваша задача — найти такие троичные числа a и b (оба длины n и без лидирующих нулей), что $a \odot b = x$ и $\max(a, b)$ — минимально возможный.

Вам нужно ответить на t независимых наборов входных данных.

Входные данные

Первая строка теста содержит одно целое число t ($1 \leq t \leq 10^4$) — количество наборов входных данных. Затем следуют t наборов входных данных. Первая строка набора содержит одно целое число n ($1 \leq n \leq 5 \cdot 10^4$) — длину числа x . Вторая строка набора содержит троичное число x , состоящее из n цифр 0, 1 и 2. Гарантируется, что первой цифрой числа x является 2. Также гарантируется, что сумма всех n по всем наборам входных данных не превосходит $5 \cdot 10^4$ ($\sum n \leq 5 \cdot 10^4$).

Выходные данные

Для каждого набора входных данных выведите ответ на него — два троичных числа a и b (каждое длины n и без лидирующих нулей) таких, что $a \odot b = x$ и $\max(a, b)$ — минимально возможное. Если есть несколько возможных ответов, вы можете вывести любой.

Пример

входные данные	Скопировать
4 5 22222 5 21211 1 2 9 220222021	
выходные данные	Скопировать
11111 11111 11000 10211 1 1 110111011 110111010	



[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

С. Экзамены

ограничение по времени на тест: 1 секунда

ограничение по памяти на тест: 256 мегабайт

Студент Валера учится на первом курсе университета. Скоро у него сессия, и ему предстоит сдать ровно n экзаменов. Валера — умный парень, поэтому он сможет сдать любой экзамен с первого раза. Кроме того, он может сдавать несколько экзаменов в один день и в любом порядке.

Согласно расписанию, экзамен по i -му предмету нужно сдать в день с номером a_i . Однако Валера договорился с каждым преподавателем, и преподаватель i -го предмета разрешил организовать досрочную сдачу своего экзамена в день b_i ($b_i < a_i$). Таким образом, Валера может сдать экзамен по i -му предмету либо в день a_i , либо в день b_i . Все преподаватели ставят запись о сдаче экзамена в зачетную книжку в день фактической сдачи экзамена и датируют эту запись числом a_i .

Валера считает, что будет достаточно странно, если записи в зачетной книжке будут идти не в порядке неубывания даты. Поэтому Валера просит вас помочь ему. Найдите минимально возможный номер дня, когда Валера сможет сдать последний экзамен, если он будет сдавать экзамены так, чтобы все записи в его зачетной книжке шли в порядке неубывания даты.

Входные данные

В первой строке записано единственное целое положительное число n ($1 \leq n \leq 5000$) — количество экзаменов, которые будет сдавать Валера.

В каждой из следующих n строк записано по два целых положительных числа через пробел a_i и b_i ($1 \leq b_i < a_i \leq 10^9$) — дата сдачи по расписанию и досрочная дата сдачи i -го экзамена соответственно.

Выходные данные

Выведите единственное целое число — минимально возможный номер дня, когда Валера сможет сдать последний экзамен, если он будет сдавать экзамены так, чтобы все записи в его зачетной книжке шли в порядке неубывания даты.

Примеры

входные данные	Скопировать
3 5 2 3 1 4 2	
выходные данные	Скопировать
2	

входные данные	Скопировать
3 6 1 5 2 4 3	
выходные данные	Скопировать
6	

Примечание

В первом примере Валера сначала сдаст экзамен по второму предмету в первый день (в зачетную книжку заносится дата сдачи по расписанию, то есть 3). На следующий день сначала он сдаст экзамен по третьему предмету (будет добавлена запись, датированная днем 4), а после него в тот же день сдаст экзамен по первому предмету (в зачетную книжку отметка будет проставлена с днем 5). Таким образом, последний экзамен Валера сдаст во второй день, и даты в зачетной книжке будут идти в порядке неубывания: 3, 4, 5.

Во втором примере Валера сначала сдаст экзамен по третьему предмету в четвертый день, после него экзамен по второму предмету в пятый день. После этого в шестой день Валера сдаст экзамен по первому предмету.

[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

С. Drazil и факториал

ограничение по времени на тест: 2 секунды
ограничение по памяти на тест: 256 мегабайт

Drazil играет в математическую игру с Varda.

Определим $F(x)$ для целого числа x как произведение факториалов его цифр. Например, $F(135) = 1! * 3! * 5! = 720$.

Сперва выбирается число a , состоящее из n цифр и содержащее не менее одной цифры, превосходящей 1. Это число может начинаться с ведущего нуля. Цель игры — найти наибольшее возможное целое число x , удовлетворяющее следующим двум условиям:

- 1. x не содержит ни цифру 0, ни цифру 1.
- 2. $F(x) = F(a)$.

Помогите друзьям найти такое число.

Входные данные

В первой строке записано целое число n ($1 \leq n \leq 15$) — количество цифр в числе a .

Во второй строке записано n цифр числа a . В числе a есть не менее одной цифры, превышающей 1. Число a может содержать ведущие нули.

Выходные данные

Выведите наибольшее возможное целое число, удовлетворяющее условиям, данным выше. В десятичной записи этого числа не должно быть нулей и единиц.

Примеры

входные данные	Скопировать
4 1234	
выходные данные	Скопировать
33222	

входные данные	Скопировать
3 555	
выходные данные	Скопировать
555	

Примечание

В первом тесте из условия $F(1234) = 1! * 2! * 3! * 4! = 288 = F(33222)$



[ЗАДАЧИ](#) [ОТΟΣЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

В. Феникс и красота

ограничение по времени на тест: 2 секунды
ограничение по памяти на тест: 256 мегабайт

Феникс любит красивые массивы. Он считает массив красивым, если все его подмассивы длины k имеют равную сумму. Подмассив массива — это последовательность подряд идущих элементов массива.

У Феникса есть массив a длины n . Он хочет вставить несколько (возможно, ноль) целых чисел в свой массив, чтобы он стал красивым. Вставляемые числа должны иметь значения от 1 по n включительно. Числа можно вставлять куда угодно (даже перед первым или после последнего элемента). Феникс **не пытается** минимизировать количество вставленных чисел.

Входные данные

Входные данные состоят из нескольких наборов. В первой строке задано целое число t ($1 \leq t \leq 50$) — количество наборов входных данных.

В первой строке каждого набора входных данных задано два целых числа n и k ($1 \leq k \leq n \leq 100$).

Во второй строке каждого набора задано n целых чисел через пробел ($1 \leq a_i \leq n$) — первоначальный массив Феникса. Массив уже может быть красивым, а может и не быть.

Выходные данные

Для каждого набора входных данных, если невозможно получить красивый массив, выведите -1. Иначе, выведите две строки.

В первой строке выведите длину красивого массива m ($n \leq m \leq 10^4$). Вам не нужно минимизировать m .

Во второй строке выведите m целых чисел через пробел ($1 \leq b_i \leq n$) — красивый массив, который Феникс сможет получить вставляя несколько (возможно, ноль) целых чисел в свой массив a . Вы можете выводить числа, которых не было в массиве a первоначально.

Если существует несколько решений, выведите любое. Гарантируется, что если мы можем сделать массив a красивым, то мы всегда сможем добиться этого при итоговой длине не более 10^4 .

Пример

входные данные	Скопировать
4 4 2 1 2 2 1 4 3 1 2 2 1 3 2 1 2 3 4 4 4 3 4 2	
выходные данные	Скопировать
5 1 2 1 2 1 4 1 2 2 1 -1 7 4 3 2 1 4 3 2	

Примечание

В первом наборе входных данных, мы можем сделать массив a красивым, если вставим число 1 в позицию 3 (между двумя числами 2). Теперь все подмассивы длины $k = 2$ будут иметь одинаковую сумму 3. Существует много других возможных решений, например:

- 2, 1, 2, 1, 2, 1
- 1, 2, 1, 2, 1, 2

Во втором наборе, массив уже красивый: все подмассивы длины $k = 3$ имеют одинаковую сумму 5.

В третьем наборе, можно показать, что невозможно вставить числа так, чтобы сделать массив a красивым.

[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

С. Максимальная медиана

ограничение по времени на тест: 2 секунды
ограничение по памяти на тест: 256 мегабайт

Вам дан массив a из n целых чисел, где n нечётно. Вы можете проделать с массивом следующую операцию:

- Выбрать один из элементов массива (например a_i) и увеличить его на 1 (то есть заменить на $a_i + 1$).

Вы хотите сделать медиану массива максимально большой, используя не более k операций.

Медианой нечётного по размеру массива называется средний элемент, если массив отсортировать по неубыванию. Например, медианой массива $[1, 5, 2, 3, 5]$ является 3.

Входные данные

Первая строка содержит два целых числа n и k ($1 \leq n \leq 2 \cdot 10^5$, n нечётно, $1 \leq k \leq 10^9$) — количество элементов в массиве и наибольшее возможное количество операций, которое можно сделать.

Вторая строка содержит n целых чисел a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$).

Выходные данные

Выведите одно целое число — наибольшую возможную медиану после всех операций.

Примеры

входные данные	Скопировать
3 2 1 3 5	
выходные данные	Скопировать
5	

входные данные	Скопировать
5 5 1 2 1 1 1	
выходные данные	Скопировать
3	

входные данные	Скопировать
7 7 4 1 2 4 3 4 4	
выходные данные	Скопировать
5	

Примечание

В первом примере можно два раза увеличить второй элемент. Тогда массив будет равен $[1, 5, 5]$ и его медиана равна 5.

Во втором примере можно один раз увеличить второе число, а также два раза увеличить третье и пятое. Тогда ответ будет равен 3.

В третьем примере можно сделать четыре операции, чтобы увеличить первый, четвёртый, шестой и седьмой элемент, тогда массив будет равен $[5, 1, 2, 5, 3, 5, 5]$ и медиана будет равна 5.

[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

С. Соответствия поворотом

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 256 мегабайт

После мистического исчезновения Ashish, каждый из его любимых учеников Ishika и Hriday, получил одну половину секретного сообщения. Эти сообщения могут быть описаны перестановками размера n . Назовем их a и b .

Напомним, что перестановка из n элементов это последовательность чисел a_1, a_2, \dots, a_n , в которой каждое число от 1 до n встречается ровно один раз.

Сообщение может быть расшифровано из конфигурации перестановок a и b , в котором количество совпадающих пар элементов максимально. Пара элементов a_i и b_j называется совпадающей, если:

- $i = j$, таким образом, у них один и тот же индекс.
- $a_i = b_j$

Его ученикам разрешается совершать следующую операцию произвольное число раз:

- выбрать число k и циклически сдвинуть одну из перестановок влево или вправо k раз.

Циклический сдвиг перестановки c влево это операция, которая присваивает $c_1 := c_2, c_2 := c_3, \dots, c_n := c_1$ одновременно. Аналогично, циклический сдвиг перестановки c вправо это операция, которая присваивает $c_1 := c_n, c_2 := c_1, \dots, c_n := c_{n-1}$ одновременно.

Помогите Ishika и Hriday найти наибольшее возможное число совпадающих пар в данных перестановках после применения описанных операций несколько (возможно, ноль) раз.

Входные данные

В первой строке записано одно целое число n ($1 \leq n \leq 2 \cdot 10^5$) — размеры массивов.
Во второй строке записаны n целых чисел a_1, a_2, \dots, a_n ($1 \leq a_i \leq n$) — элементы первой перестановки.
В третьей строке записаны n целых чисел b_1, b_2, \dots, b_n ($1 \leq b_i \leq n$) — элементы второй перестановки.

Выходные данные

Выведите наибольшее возможное число совпадающих пар в данных перестановках после применения описанных операций несколько (возможно, ноль) раз.

Примеры

входные данные	Скопировать
5 1 2 3 4 5 2 3 4 5 1	
выходные данные	Скопировать
5	
входные данные	Скопировать
5 5 4 3 2 1 1 2 3 4 5	
выходные данные	Скопировать
1	
входные данные	Скопировать
4 1 3 2 4 4 2 3 1	
выходные данные	Скопировать
2	

Примечание

В первом примере можно сдвинуть b направо на $k = 1$. Получившиеся перестановки будут $\{1, 2, 3, 4, 5\}$ и $\{1, 2, 3, 4, 5\}$.

Во втором примере не требуется совершать никаких операций. По всем возможным сдвигам a и b , число совпадающих пар не будет превышать 1.

В третьем примере можно сдвинуть b влево на $k = 1$. Получившиеся перестановки будут $\{1, 3, 2, 4\}$ и $\{2, 3, 1, 4\}$. Позиции 2 и 4 будут являться совпадающей парой. По всем возможным циклическим сдвигам a и b , количество совпадающих пар не будет превышать 2.

[Codeforces](#) (c) Copyright 2010-2025 Михаил Мирзаянов

Соревнования по программированию 2.0

Время на сервере: 07.02.2025 21:24:08^{UTC+5} (k2).

Мобильная версия, переключиться на [десктопную](#).

[Privacy Policy](#)

При поддержке



ИТМО

[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

D. Испорченный массив

ограничение по времени на тест: 2 секунды

ограничение по памяти на тест: 256 мегабайт

Вам дано число n и массив b_1, b_2, \dots, b_{n+2} , полученный согласно следующему алгоритму:

- был задан некоторый массив a_1, a_2, \dots, a_n ;
- массив a записали в массив b , т.е. $b_i = a_i$ ($1 \leq i \leq n$);
- $(n + 1)$ -м элементом массива b записали сумму чисел в массиве a , т.е. $b_{n+1} = a_1 + a_2 + \dots + a_n$;
- $(n + 2)$ -м элементом массива b записали некоторое число x ($1 \leq x \leq 10^9$), т.е. $b_{n+2} = x$;
- массив b был перемешан.

Например, массив $b = [2, 3, 7, 12, 2]$ мог быть получен следующими способами:

- $a = [2, 2, 3]$ и $x = 12$;
- $a = [3, 2, 7]$ и $x = 2$.

Для заданного массива b найдите любой массив a , который мог быть задан изначально.

Входные данные

В первой строке содержится одно целое число t ($1 \leq t \leq 10^4$). Далее следуют t наборов входных данных.

Первая строка каждого набора содержит одно целое число n ($1 \leq n \leq 2 \cdot 10^5$).

Вторая строка каждого набора содержит $n + 2$ целых числа b_1, b_2, \dots, b_{n+2} ($1 \leq b_i \leq 10^9$).

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

Выходные данные

Для каждого набора входных данных выведите:

- «-1», если массив b не мог быть получен ни по какому массиву a ;
- n целых чисел a_1, a_2, \dots, a_n , иначе.

Если существует несколько массивов a , то можете выводить любой.

Пример

входные данные	Скопировать
4 3 2 3 7 12 2 4 9 1 7 1 6 5 5 18 2 2 3 2 9 2 3 2 6 9 2 1	
выходные данные	Скопировать
2 3 7 -1 2 2 2 3 9 1 2 6	