

Технокубок 2019 - Отборочный Раунд 4

А. Право-левое шифрование

1 секунда, 256 мегабайт

Поликарп обожает шифры. Недавно он изобрёл свой собственный, который назвал право-левым.

Право-левое шифрование используется для шифрования строк. Чтобы зашифровать строку $s = s_1 s_2 \dots s_n$ Поликарп выполняет следующие шаги:

- он выписывает s_1 ,
- к текущему результату приписывает справа s_2 ,
- к текущему результату приписывает слева s_3 ,
- к текущему результату приписывает справа s_4 ,
- к текущему результату приписывает слева s_5 ,
- и так далее, пока не будут обработаны все символы строки.

Например, если $s = \text{"techno"}$, то процесс шифрования будет выглядеть так: $t \rightarrow te \rightarrow cte \rightarrow cteh \rightarrow ncteh \rightarrow ncteho$. Таким образом, для $s = \text{"techno"}$ результат шифрования равен "ncteho" .

Дана строка t — результат шифрования некоторой строки s . Ваша задача расшифровать, то есть найти строку s .

Входные данные

Единственная строка входных данных содержит t — результат шифрования некоторой строки s . Заданная строка содержит только строчные буквы латинского алфавита, длина t — от 1 до 50 включительно.

Выходные данные

Выведите такую строку s , которая после шифрования равна t .

Входные данные

ncteho

Выходные данные

techno

Входные данные

erfdcoeocs

Выходные данные

codeforces

Входные данные

z

Выходные данные

z

В. Div на Mod

1 секунда, 256 мегабайт

Вася любит решать уравнения. Сегодня он хочет решить уравнение $(x \operatorname{div} k) \cdot (x \operatorname{mod} k) = n$, где div и mod означают целочисленное деление и операцию взятия по модулю (см. Примечания ниже для точного определения). В этом уравнении k и n являются целыми положительными параметрами, а x — неизвестным целым положительным числом. Если решений несколько, то Вася хочет найти наименьшее из возможных x . Вы можете ему помочь?

Входные данные

Первая строка содержит два целых числа n и k ($1 \leq n \leq 10^6$, $2 \leq k \leq 1000$).

Выходные данные

Выведите единственное целое число x — наименьшее положительное целое решение $(x \operatorname{div} k) \cdot (x \operatorname{mod} k) = n$. Гарантируется, что это уравнение имеет хотя бы одно целое положительное решение.

входные данные
6 3
выходные данные
11

входные данные
1 2
выходные данные
3

входные данные
4 6
выходные данные
10

Результат целочисленного деления $a \operatorname{div} b$ равен наибольшему целому числу c такому, что $b \cdot c \leq a$. $a \operatorname{mod} b$ — единственное целое число c такое, что $0 \leq c < b$ и $a - c$ делится на b .

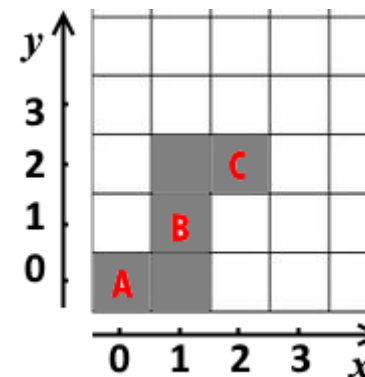
В первом примере $11 \operatorname{div} 3 = 3$ и $11 \operatorname{mod} 3 = 2$. Поскольку $3 \cdot 2 = 6$, то $x = 11$ это решение уравнения $(x \operatorname{div} 3) \cdot (x \operatorname{mod} 3) = 6$. Можно заметить, что 19 — это единственное другое решение этого уравнения, поэтому 11 будет наименьшим.

С. Соедините тройку

1 секунда, 256 мегабайт

Национальный лес Квадрляндии был разделен на одинаковые квадратные участки размером 1×1 , выровненные по направлениям с севера на юг и с востока на запад. Каждый участок может быть однозначно задан целочисленными Декартовыми координатами (x, y) своего юго-западного угла.

Три друга, Алиса, Боб и Чарли собираются купить три отдельных участка земли A, B, C в лесу. Изначально все участки в лесу (включая участки A, B, C) покрыты деревьями. Друзья хотят навещать друг друга, поэтому они хотят очистить некоторые участки от деревьев. После очистки, можно будет добраться до любого из участков A, B, C от любого другого, перемещаясь по соседним очищенным участкам. Два земельных участка являются соседними, если они имеют общую сторону.



Например, $A = (0, 0)$, $B = (1, 1)$, $C = (2, 2)$. Необходимо очистить как минимум 5 участков, чтобы соединить друзей. Эти участки выделены серым цветом. Конечно, друзья не хотят слишком напрягаться. Помогите им найти наименьшее количество участков, которые они должны очистить от деревьев.

Входные данные

Первая строка содержит два целых числа x_A и y_A — координаты участка A ($0 \leq x_A, y_A \leq 1000$). Следующие две строки описывают координаты (x_B, y_B) и (x_C, y_C) участков B и C , соответственно, в таком же формате ($0 \leq x_B, y_B, x_C, y_C \leq 1000$). Гарантируется, что все три участка различны.

Выходные данные

В первой строке выведите одно целое число k — наименьшее количество участков, которые необходимо очистить от деревьев. Следующие k строк должны содержать координаты всех участков, которые необходимо очистить. Все k участков должны быть различны. Участки можно вывести в любом порядке.

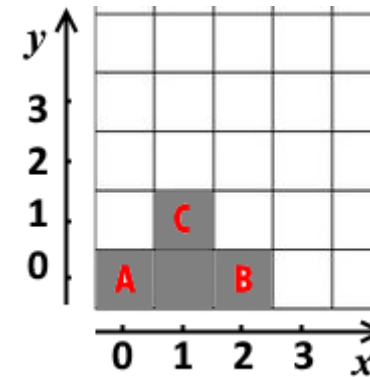
Если решений несколько, выведите любое из них.

входные данные
0 0 1 1 2 2
выходные данные
5 0 0 1 0 1 1 1 2 2 2

входные данные
0 0 2 0 1 1
выходные данные
4 0 0 1 0 1 1 2 0

Первый пример показан на картинке в условии задачи.

Второй пример изображен на этой картинке:



D. Дерево минимального диаметра

1 секунда, 256 мегабайт

Вам дано дерево (неориентированный связный граф без циклов) и целое число s .

Ваня хочет как-то поставить веса на всех ребрах дерева, так что все веса неотрицательные вещественные числа, которые в сумме дают s . При этом он хочет, чтобы диаметр получившегося после расстановки весов дерева был как можно меньше. Диаметром будем называть максимальную сумму весов ребер, лежащих на пути между какими-то двумя вершинами дерева.

Найдите минимальный возможный диаметр, который мог у него получиться.

Входные данные

В первой строке даны два целых числа n и s ($2 \leq n \leq 10^5$, $1 \leq s \leq 10^9$) — количество вершин в дереве и сумма весов ребер.

В каждой из следующих $n - 1$ строк через пробел записаны по два целых числа a_i и b_i ($1 \leq a_i, b_i \leq n$, $a_i \neq b_i$) — номера вершин, соединенных ребром.

Гарантируется, что заданные ребра задают дерево.

Выходные данные

Выведите минимальный диаметр дерева, который мог получиться при какой-то расстановке неотрицательных весов на ребра этого дерева, которые в сумме дают s .

Ваш ответ будет считаться правильным, если его абсолютная или относительная ошибка не будет превосходить 10^{-6} .

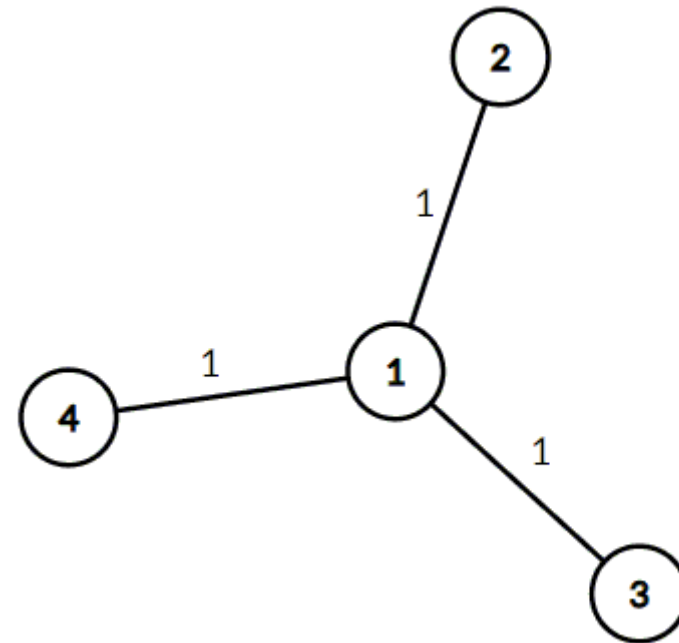
А именно: пусть ваш ответ равен a , а ответ жюри b . Ваш ответ будет засчитан, если $\frac{|a-b|}{\max(1,b)} \leq 10^{-6}$.

Входные данные
4 3 1 2 1 3 1 4
Выходные данные
2.000000000000000000

Входные данные
6 1 2 1 2 3 2 5 5 4 5 6
Выходные данные
0.500000000000000000

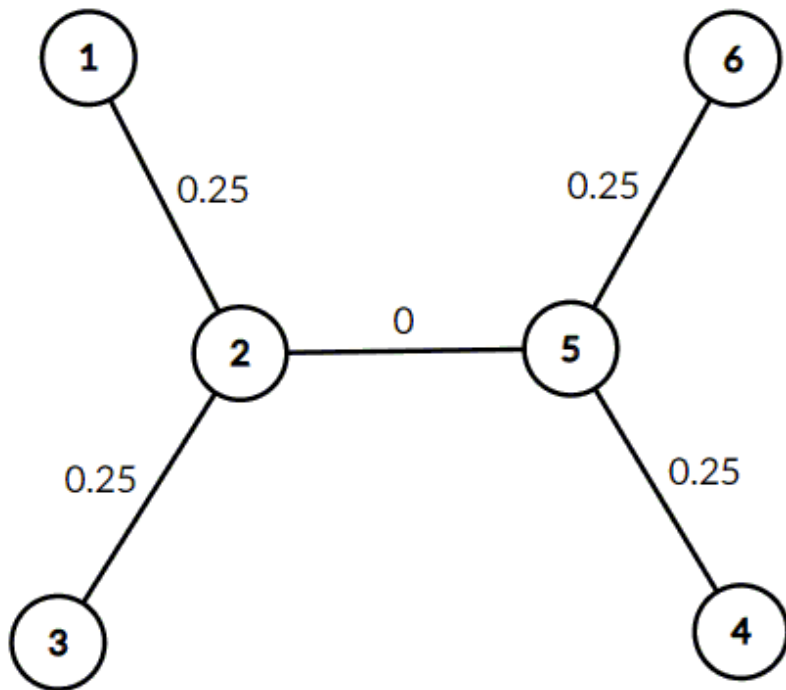
Входные данные
5 5 1 2 2 3 3 4 3 5
Выходные данные
3.3333333333333333

В первом примере необходимо расставить веса так:



Легко видеть, что диаметр такого дерева будет равен 2. Можно доказать, что это минимально возможный диаметр, который можно получить, как-то расставив веса.

Во втором примере необходимо расставить веса так:



Е. Вася и паттерны

5 секунд, 256 мегабайт

У Васи есть три строки s , a и b , состоящие из первых k букв латинского алфавита.

Назовем паттерном строку длины k , где каждая из первых k букв латинского алфавита встречается ровно один раз (таким образом, существует ровно $k!$ различных паттернов). Применение паттерна p к строке s — это замена каждой буквы строки s на p_i , где i — это порядковый номер этой буквы в латинском алфавите. Например, применив паттерн «bdca» к строке «aabccd» мы получим строку «bbdcca».

Васю интересует, существует ли такой паттерн, применив который к строке s , он получит строку, лексикографически больше либо равную строки a и лексикографически меньше либо равную строки b .

Если существует несколько подходящих паттернов, разрешается вывести любой.

Строка a лексикографически меньше строки b , если существует такое i ($1 \leq i \leq n$), что $a_i < b_i$, а для любого j ($1 \leq j < i$) $a_j = b_j$.

В этой задаче вам необходимо ответить на t **независимых** тестовых наборов.

Входные данные

Первая строка входных данных содержит одно целое число t ($1 \leq t \leq 10^6$) — количество тестовых наборов.

Во **взломах** можно использовать только $t = 1$.

Каждая из следующих t строк содержит описание тестового набора в следующем формате:

Первая строка тестового набора содержит целое число k ($1 \leq k \leq 26$) — длина необходимого паттерна.

Вторая строка тестового набора содержит строку s ($1 \leq |s| \leq 10^6$).

Третья строка тестового набора содержит строку a .

Четвертая строка тестового набора содержит строку b .

Строки s , a и b имеют одинаковую длину ($|s| = |a| = |b|$) и состоят из первых k букв латинского алфавита, все буквы строчные.

Гарантируется, что строка a лексикографически меньше либо равна строке b .

Также гарантируется, что сумма длин строк по всем тестовым наборам не превосходит $3 \cdot 10^6$.

Выходные данные

На каждый тестовый набор выведите ответ в следующем формате:

Если не существует подходящего паттерна, то в первой строке выведите «NO».

Иначе в первой строке выведите «YES», а во второй сам паттерн (k строчных букв, каждая из первых k букв латинского алфавита должна встретиться ровно один раз).

Если существует несколько подходящих паттернов — разрешается вывести любой.

ВХОДНЫЕ ДАННЫЕ
2 4 bbcb aada aada 3 abc bbb bbb
ВЫХОДНЫЕ ДАННЫЕ
YES badc NO

Г. Победитель в Камень-Ножницы-Бумагу

2 секунды, 256 мегабайт

n игроков собираются провести турнир по игре «камень-ножницы-бумага». Как вы наверное знаете, при игре один на один в камень-ножницы-бумагу каждый из игроков выбирает себе предмет независимо друг от друга. Затем результат определяется в зависимости от выбранных предметов: "бумага" побеждает "камень", "камень" побеждает "ножницы", "ножницы" побеждают "бумагу", и два одинаковых предмета приводят к ничьей.

В начале турнира все игроки будут стоять в ряд. Они пронумерованы по возрастанию от 1 до n от самого левого игрока к самому правому. Каждый игрок имеет заранее выбранный предмет, который он будет использовать в каждой игре на протяжении всего турнира. Вот как будет проводиться турнир:

- Если остался только один игрок, он объявляется победителем.
- Иначе, в ряду выбираются произвольно два соседних игрока. Они играют следующий матч. Проигравший игрок выбывает из турнира и покидает свое место в ряду (его бывшие соседи становятся соседями). Если игра завершилась ничьей, проигравший игрок определяется подбрасыванием монеты.

Организаторы знают о предпочтительных предметах всех игроков. Они хотят узнать общее количество игроков, у которых есть шанс стать победителем турнира (это значит, что есть подходящий способ выбрать порядок игр и манипулировать бросками монет). Тем не менее, некоторые игроки все еще оптимизируют свою стратегию и могут сообщить организаторам о своих новых предпочтительных предметах. Можете ли вы найти количество возможных победителей после каждого такого запроса?

Входные данные

Первая строка содержит два целых числа n и q — количество игроков и запросов изменения предпочтительного предмета ($1 \leq n \leq 2 \cdot 10^5, 0 \leq q \leq 2 \cdot 10^5$).

Вторая строка содержит строку из n символов. i -й из этих символов "R", "P", или "S", если игрок с номером i собирается играть перед всеми изменениями, используя "камень", "бумагу", или "ножницы", соответственно.

Следующие q строк описывают изменения. j -я из этих строк содержит целое число p_j и символ c_j , означающий, что игрок p_j будет использовать в играх предмет, обозначенный символом c_j , начиная с этого момента ($1 \leq p_j \leq n$).

Выходные данные

Выведите $q + 1$ целое число r_0, \dots, r_q , где r_k это количество игроков, которые могут стать победителями после применения первых k изменений.

входные данные
3 5 RPS 1 S 2 R 3 P 1 P 2 P
выходные данные
2 2 1 2 2 3

G. Красивая раскраска

4 секунды, 256 мегабайт

Петя является коллекционером *красивых* матриц.

Матрица размера $n \times n$ является *красивой*, если:

- Все элементы матрицы являются целыми числами от 1 до n ;
- В строке все элементы различные;
- В столбце нет двух соседних одинаковых элементов.

Сегодня Петя купил *красивую* матрицу a размера $n \times n$ и хочет определить ее редкость.

Редкость матрицы определяется, как ее номер в списке всех *красивых* матриц размера $n \times n$, отсортированных в лексикографическом порядке. Сравнение матриц происходит построчно. (Нумерация начинается с **нуля**).

Так как *красивых* матриц очень много, то Пете будет достаточно узнать редкость матрицы a по модулю 998 244 353.

Входные данные

В первой строке записано целое число n ($1 \leq n \leq 2000$) — размеры матрицы a .

В последующих n строках записано по n целых чисел $a_{i,j}$ ($1 \leq a_{i,j} \leq n$) — элементы матрицы a .

Гарантируется, что матрица a удовлетворяет свойствам *красивой* матрицы.

Выходные данные

Выведите одно целое число — редкость матрицы a по модулю 998 244 353.

входные данные
2 2 1 1 2
выходные данные
1

входные данные
3 1 2 3 2 3 1 3 1 2
выходные данные
1

входные данные
3 1 2 3 3 1 2 2 3 1

выходные данные

3

Для матриц размера 2×2 существует всего 2 *красивые* матрицы:

1 2	2 1
2 1	1 2

Красивых матриц 3×3 довольно много, вот первые 5 в лексикографическом порядке:

1 2 3	1 2 3	1 2 3	1 2 3	1 3 2
2 3 1	2 3 1	3 1 2	3 1 2	2 1 3
1 2 3	3 1 2	1 2 3	2 3 1	1 3 2