

Задача А. Пути в дереве

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 10 секунд
Ограничение по памяти: 512 мегабайт

Дано дерево из n вершин. Найдите для каждого d от 1 до $n - 1$ число путей длины d .

Формат входных данных

Первая строка содержит n — число вершин дерева ($1 \leq n \leq 50000$).

Следующие $n - 1$ строк содержат ребра дерева. Каждое ребро описывается парой чисел v_i, u_i — концы ребра ($1 \leq v_i, u_i \leq n$).

Формат выходных данных

Выведите $n - 1$ число: i -е число — число путей длины i .

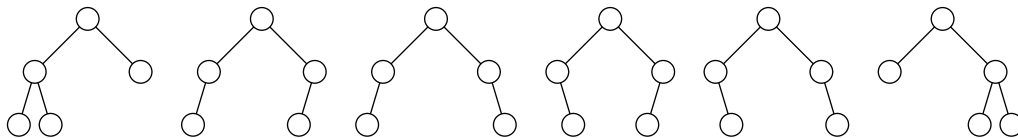
Примеры

стандартный ввод	стандартный вывод
3 1 2 2 3	2 1
9 3 2 4 2 1 2 5 1 1 6 7 6 6 8 8 9	8 10 10 6 2 0 0 0

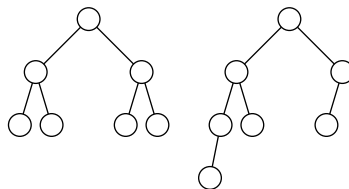
Задача В. AVL

Имя входного файла: `avl.in`
 Имя выходного файла: `avl.out`
 Ограничение по времени: 2 секунды
 Ограничение по памяти: 256 мегабайт

AVL деревья, придуманные российскими учёными Адельсон-Вельским и Ландисом, являются примером сбалансированного бинарного дерева поиска. В терминологии AVL, подвешенное бинарное дерево называется сбалансированным, если для каждой вершины высоты её левого и правого поддеревьев отличаются не более, чем на один. Такое дерево, собственно, и называется AVL-деревом. Разумеется, существует далеко не единственное AVL-дерево при фиксированном числе вершин. К примеру, существует шесть AVL-деревьев с пятью вершинами, они изображены на рисунке ниже.



Деревья с одинаковым числом вершин могут иметь разную высоту, к примеру, на рисунке снизу нарисовано два дерева с семью вершинами, которые имеют высоты 2 и 3, соответственно.



Вам даны два числа — N и H , требуется найти число AVL-деревьев, которые состоят из N вершин и имеют высоту H . Поскольку их число довольно велико, выведите искомое количество по модулю 786 433.

Формат входных данных

Единственная строка входного файла содержит два числа — N и H ($1 \leq N \leq 65\,535$, $0 \leq H \leq 15$).

Формат выходных данных

Выведите единственное число — количество AVL деревьев с N вершинами высоты H , по модулю 786 433.

Примеры

<code>avl.in</code>	<code>avl.out</code>
7 3	16

Замечание

786 433 простое число, и $786\,433 = 3 \cdot 2^{18} + 1$.

Задача С. Дуэль

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Двое дуэлянтов решили выбрать в качестве места проведения поединка тёмную аллею. Вдоль этой аллеи растёт n деревьев и кустов. Расстояние между соседними объектами равно одному метру. Дуэль решили проводить по следующим правилам. Некоторое дерево выбирается в качестве стартовой точки. Затем два дерева, находящихся на одинаковом расстоянии от исходного, отмечаются как места для стрельбы. Дуэлянты начинают движение от стартовой точки в противоположных направлениях. Когда соперники достигают отмеченных деревьев, они разворачиваются и начинают стрелять друг в друга.

Дана схема расположения деревьев вдоль аллеи. Требуется определить количество способов выбрать стартовую точку и места для стрельбы согласно правилам дуэли.

Формат входных данных

Во входном файле содержится одна строка, состоящая из символов '0' и '1' — схема аллеи. Деревья обозначаются символом '1', кусты — символом '0'. Длина строки не превосходит 100 000 символов.

Формат выходных данных

Выведите количество способов выбрать стартовую точку и места для стрельбы согласно правилам дуэли.

Примеры

<code>stdin</code>	<code>stdout</code>
101010101	4
101001	0

Замечание

В первом примере возможны следующие конфигурации дуэли (стартовое дерево и деревья для стрельбы выделены жирным шрифтом): **101010101**, **101010101**, **101010101** и **101010101**.

Задача D. Задача для второклассника

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Вам даны два числа. Необходимо найти их произведение.

Формат входных данных

Входные данные состоят из двух строк, на каждой из которых находится целое одно **целое** число, длина которого не превосходит двухсот пятидесяти тысяч символов.

Формат выходных данных

Выведите произведение данных чисел.

Примеры

<code>stdin</code>	<code>stdout</code>
2	4
2	

Задача Е. Супер-счастливые билеты

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

Каждому школьнику параллели нужно выдать билетик. На каждом билетике должна быть написана последовательность из n цифр, где n — четное положительное число.

Даниил считает билет супер-счастливым, если выполняются два свойства:

- Сумма первых $\frac{n}{2}$ цифр равна сумме последних $\frac{n}{2}$ цифр.
- Сумма цифр на четных позициях равна сумме цифр на нечетных позициях.

Помогите Даниилу посчитать количество различных супер-счастливых билетиков по модулю 998 244 353.

Формат входных данных

В первой строке дано одно число n — длина билета ($2 \leq n \leq 200\,000$). Гарантируется, что n чётное.

Формат выходных данных

Выведите количество супер-счастливых билетиков длины n по модулю 998 244 353.

Примеры

стандартный ввод	стандартный вывод
2	10
8	448900

Задача F. Блэк & Уайт

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	5 секунд
Ограничение по памяти:	512 мегабайт

В новом регионе всего $n + 1$ городов, соединенных двусторонними дорогами. Причем, n из этих городов расположены на окружности, а один город является столицей и расположен в центре. Прономеруем города на окружности от 1 до n в порядке обхода, и назовем столицу номер $n + 1$. Каждая дорога либо соединяет столицу с городом на окружности, либо соединяет два соседних города на окружности. Иными словами, дорога соединяет города $(n + 1)$ и v или города v и $(v \bmod n + 1)$, где $v \in [1, n]$.

Каждая дорога контролируется одной из конкурирующих банд: либо бандой Уайта, либо бандой Блэка. Вы хотите выбрать и обезопасить минимальное количество дорог, по которым можно было бы добраться от любого города до любого другого. Другими словами, требуется выбрать остовное дерево в графе городов. Можно доказать, что любое остовное дерево будет содержать ровно n дорог.

Для того, чтобы обезопасить дорогу, нужно договориться с главарём банды, контролирующей эту дорогу. Пока неизвестно, кто из Уайта и Блэка окажется сговорчивее, поэтому для каждого целого $k \in [0, n]$ надо посчитать количество способов выбрать остовное дерево так, чтобы ровно k из выбранных дорог контролировались бандой Уайта, а $n - k$ оставшихся дорог — бандой Блэка. Так как ответы могут быть большими, посчитайте их по модулю 998 244 353.

Формат входных данных

В первой строке дано одно целое число n ($3 \leq n \leq 50\,000$).

Во второй строке дана строка s , состоящая из n символов, описывающая дороги между городами на окружности. Если s_i равно «-», дорога между городами i и $(i \bmod n + 1)$ отсутствует, если «W» — дорога контролируется бандой Уайта, и если «B» — бандой Блэка.

В третьей строке дана строка t , состоящая из n символов, описывающая дороги между столицей и городами на окружности. Если t_i равно «-», дорога между столицей и городом i отсутствует, если «W» — дорога контролируется бандой Уайта, и если «B» — бандой Блэка.

Формат выходных данных

Выведите $n + 1$ целое число a_k — k -е из них должно равняться количеству остовных деревьев графа городов, в которых ровно k дорог контролируются бандой Уайта.

Примеры

стандартный ввод	стандартный вывод
3 --- WBW	0 0 1 0
3 WWW BBB	1 6 9 0
5 BWB-B WB-W-	0 2 6 3 0 0
4 ---- ----	0 0 0 0 0

Задача G. Множество делителей

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 5 секунд
Ограничение по памяти: 256 мегабайт

Вам задано число x , представленное в виде произведения n своих простых делителей $p_1 \cdot p_2 \cdot \dots \cdot p_n$. Пусть S — это множество всех положительных целых делителей числа x (включая 1 и само число x).

Назовем множество (set) целых чисел D хорошим тогда (и только тогда), когда не существует пары $a \in D, b \in D$ таких, что $a \neq b$ и a делит b .

Найдите хорошее подмножество S с максимально возможным размером. Так как ответ может быть очень большим, выведите размер подмножества по модулю 998244353.

Формат входных данных

В первой строке задано единственное число n ($1 \leq n \leq 2 \cdot 10^5$) — количество простых делителей в представлении числа x .

Во второй строке заданы n простых чисел $p_1, p_2, \dots, p_n$ ($2 \leq p_i \leq 3 \cdot 10^6$) — разложение числа x на простые.

Формат выходных данных

Выведите максимально возможный размер хорошего подмножества по модулю 998244353.

Примеры

стандартный ввод	стандартный вывод
3 29999999 43 2999957	3
6 2 3 2 3 2 2	3

Замечание

В первом примере $x = 2999999 \cdot 43 \cdot 2999957$ и одним из его максимальных хороших подмножеств является $\{43, 2999957, 2999999\}$.

Во втором примере $x = 2 \cdot 3 \cdot 2 \cdot 3 \cdot 2 \cdot 2 = 144$ и одним из его максимальных хороших подмножеств является $\{9, 12, 16\}$.

Задача Н. Системы счисления

Имя входного файла: `radix.in`
Имя выходного файла: `radix.out`
Ограничение по времени: 0.75 секунд
Ограничение по памяти: 256 мегабайт

Переведите число из системы счисления a в систему счисления b .

Формат входных данных

В первой строке задана начальная система счисления a ($2 \leq a \leq 10$). Во второй строке задано само число, его длина не превышает 100 000 символов. В третьей строке записана система счисления b , в которую нужно перевести число ($2 \leq b \leq 10$).

Формат выходных данных

В единственной строке выведите заданное число в системе счисления b .

Примеры

radix.in	radix.out
3	27
1000	
10	

Задача I. Продолжи последовательность

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	256 мегабайт

Наверняка вы не раз видели головоломки вида «Дана последовательность, найдите следующее число». В детстве они кажутся логичными, но потом приходит понимание, что можно написать любое число, а затем объяснить это какой-нибудь хитрой конструкцией.

Вам мы предлагаем продолжить последовательность «наиболее простым способом». Всё ещё недостаточно строго? Дадим точное определение.

Сложностью последовательности $a_1, a_2, \dots, a_n$ называется минимальное такое d , что существует многочлен p степени d такой, что $p(x) = a_x \bmod 998244353$ для всех x от 1 до n . В рамках этой задачи считайте, что степень многочлена $p(x) = 0$ равна -1 .

Вам дана последовательность $a_1, a_2, \dots, a_n$ длины n , вам нужно построить последовательность $b_1, b_2, \dots, b_{n+m}$ длины $n + m$ такую, что:

- $0 \leq b_i < 998244353$ для всех i от 1 до $n + m$
- $a_i = b_i$ для всех i от 1 до n
- Сложность последовательности b как можно меньше.

Формат входных данных

В первой строке записано два целых числа n и m ($1 \leq n \leq 10^5$, $1 \leq m \leq 8 \cdot 10^5$).

Во второй строке записаны n целых чисел a_i — исходная последовательность ($0 \leq a_i < 998244353$).

Формат выходных данных

Выведите m чисел $b_{n+1}, b_{n+2}, \dots, b_{n+m}$. Разделяйте числа пробелами.

Примеры

стандартный ввод	стандартный вывод
5 10 1 4 9 16 25	36 49 64 81 100 121 144 169 196 225
3 3 0 0 0	0 0 0
5 10 1 2 4 8 16	31 57 99 163 256 386 562 794 1093 1471
3 1 2 1 0	998244352