

Задача А. Задача для второклассника

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

стандартный ввод	стандартный вывод
2 2	4
1 -1	-1
-1 -239	239
0 -1	0

Задача В. ФФТ по известному модулю

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

Предлагается в этой задаче реализовать перемножение двух многочленов по известному модулю $MOD = 998\,244\,353$. Используя написанную вами в этой задаче функцию, вы сможете сдавать без проблем большинство задач по комбинаторике, в которой нужно ФФТ.

Формат входных данных

В первой строке находится целое число n ($0 \leq n \leq 18$).

Во второй строке находится 2^n целых чисел $a_0, a_1, \dots, a_{2^n-1}$ ($0 \leq a_i < MOD - 1$). Первый многочлен для перемножения это $A(x) = \sum_{i=0}^{2^n-1} a_i x^i$.

Во второй строке находится 2^n целых чисел $b_0, b_1, \dots, b_{2^n-1}$ ($0 \leq b_i < MOD - 1$). Первый многочлен для перемножения это $B(x) = \sum_{i=0}^{2^n-1} b_i x^i$.

Формат выходных данных

Пусть многочлен $C(x) = A(x) \cdot B(x)$. Все коэффициенты при перемножении берутся по модулю MOD . Тогда напишем, что $C(x) = \sum_{i=0}^{2^{n+1}-1} c_i x^i$. Выведите $c_0, c_1, \dots, c_{2^{n+1}-1}$.

Пример

стандартный ввод	стандартный вывод
1 1 1 1 1	1 2 1 0

Задача С. Количество путей

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

Дано дерево из n вершин. Для каждого $d = 1 \dots n - 1$ найдите количество путей длины d .

Формат входных данных

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

Следующие $n - 1$ строк содержат пары чисел u_i, v_i ($1 \leq u_i, v_i \leq n$), описывающие рёбра дерева.

Формат выходных данных

Выведите $n - 1$ число, где i -е — количество путей длины i .

Пример

стандартный ввод	стандартный вывод
3	2
1 2	1
2 3	

Задача D. HEX-Hell и сломанная строка

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Серёжа потерял место, где в редакторе шестнадцатеричных кодов HEX-Hell находилась его строка. И, так как он большой молодец, саму строку он сломал.

Напоминаем, что один байт — две шестнадцатеричных цифры из диапазона $[0-9A-F]$. Последовательность кодов в редакторе в данный момент имеет длину не более 125 000 байт (т.е. в ней не более 250 000 символов, и она имеет четную длину).

А также, у Вас есть серёжина битовая строка из нулей и единиц, «поломанная» в некоторых местах (некоторые биты будут заменены на знаки ?).

Пожалуйста, для каждого возможного начала этой строки в редакторе выведите количество совпадающих нулей и единиц. При этом можно считать, что знаки вопроса совпадают с чем угодно! По этой информации горе-Серёжа разберется со своими строками сам. Только найдите количество совпадений!

Формат входных данных

Во входном файле две непустых строки. Первая состоит из символов от 0 до 9 и от A до F. Количество таких символов чётно и не более 250 000.

Вторая строка состоит из нулей, единиц и знаков вопроса. Её длина кратна восьми (она тоже задаёт последовательность байт) и не превосходит четырех длин первой строки (она помещается в редактор).

Формат выходных данных

Если длина (в символах) первой строки n , а второй k , то Вам необходимо вывести в первой строке выходного файла $n/2 - k/8 + 1$ целых чисел — количество совпадающих бит при прикладывании второй строки к некоторому месту первой строки.

Прикладывания упорядочены естественным образом — слева направо.

Пример

стандартный ввод	стандартный вывод
B2D6 1011?010	8 5

Замечание

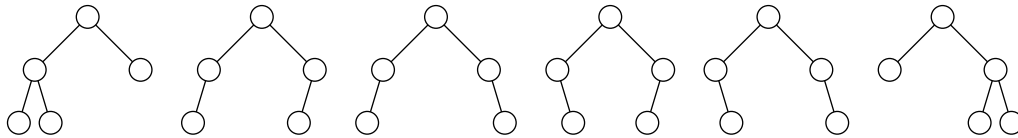
B2D6 в двоичном виде это 10110010 11010110

При прикладывании шаблона 1011?010 к первому байту 10110010 получаем восемь совпадений, а ко второму 11010110 — пять.

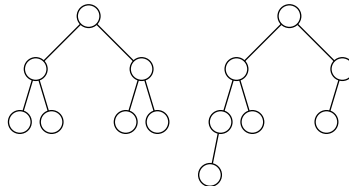
Задача Е. AVL

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.5 секунд
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.

Пример

стандартный ввод	стандартный вывод
7 3	16

Замечание

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

Задача F. Название

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Вы знаете разницу между отелем и мотелем? Вы правы, разница в количестве мух, живущих там. Петя владелец одного из самых популярных мотелей в Берляндии, но его мама настаивает на том, чтобы он превратил свой мотель в отель. Именно поэтому она подарила Пете мухобойку в виде многоугольника из k вершин.

Подойдя к окну Петя увидел n мух. Так как Петя и мухи не обидит, он хочет узнать количество способов ударить мухобойкой, не покалечив ни одной мухи.

Окно представляет из себя прямоугольник, нижний левый угол которого находится в центре координатной системы. После удара Пети все вершины мухобойки должны находиться в целых координатах и не должны вылезать за границы окна. Муха покалечена, если она будет находиться под мухобойкой либо на её границах.

Формат входных данных

Первая строка входных данных содержит числа x_p , y_p и n ($1 \leq x_p, y_p \leq 500$, $0 \leq n \leq x_p \cdot y_p$) — координаты правого верхнего угла окна и количество мух на ней, соответственно.

Следующие n строк содержат по два числа x и y ($0 < x < x_p$, $0 < y < y_p$) — координаты мух на окне.

Следующая строка содержит число k ($3 \leq k \leq 10\,000$) — количество вершин у мухобойки. В следующих k строках даются x_i и y_i ($-10^9 \leq x_i, y_i \leq 10^9$) — координаты i -й вершины мухобойки. Вершины заданы в порядке обхода, так что соседние вершины и первая и последняя вершины соединены прямой линией.

Формат выходных данных

Выведите количество способов ударить мухобойкой так, чтобы ни одна муха не пострадала.

Система оценки

Подзадача 1 (30 баллов): $x_p \cdot y_p \cdot n \cdot k \leq 3 \cdot 10^7$ и $x_p, y_p \leq 100$

Подзадача 2 (20 баллов): $x_p, y_p \leq 100$

Подзадача 3 (20 баллов): мухобойка является прямоугольником со сторонами параллельными осям координат

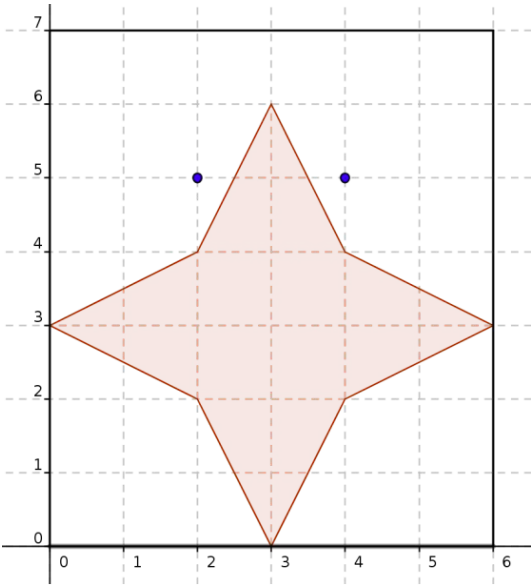
Подзадача 4 (30 баллов): нет дополнительных ограничений

Примеры

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

Замечание

Пояснение к третьему примеру:



Задача G. Дорешайте задачу

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

Лёша очень долго решал задачу и в ходе решения получил два массива s и r длины n . Чтобы дорешать задачу ему надо вычислить массив dp длины n . Для вычисления этого массива надо перебрать i от 0 до $n - 1$ и посчитать dp_i по формуле:

$$dp_i = c_i + \sum_{j=1}^i dp_{i-j} \cdot r_j$$

К сожалению, Лёша так и не успел посчитать массив dp , но успел найти массивы s и r , поэтому он просит вас найти массив dp , чтобы дорешать задачу. Так как Лёша решал «современную» задачу, все вычисления в ней проводились по модулю $M = 998244353$.

Формат входных данных

Первая строка содержит единственное целое число n ($1 \leq n \leq 2 \cdot 10^5$) — размер массивов r и s .

Вторая строка содержит n целых чисел $c_0, c_1, \dots, c_{n-1}$ ($0 \leq c_i < M$) — элементы массива s .

Третья строка содержит n целых чисел $r_0, r_1, \dots, r_{n-1}$ ($0 \leq r_i < M$) — элементы массива r .
Гарантируется, что $r_0 = 0$.

Формат выходных данных

Выведите n целых чисел, i -е из которых должно быть равно dp_i ($0 \leq dp_i < M$).

Примеры

стандартный ввод	стандартный вывод
4 1 1 1 1 0 1 1 1	1 2 4 8
5 1 2 3 4 5 0 4 3 2 1	1 6 30 144 684
7 1 3 3 7 2 2 8 0 2 2 8 2 2 8	1 5 15 55 184 612 2088

Задача Н. Странное FFT

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

Егору было лень придумывать легенду, поэтому ловите формальное условие.
Даны две последовательности A и B размеров n и m соответственно (нумерация с нуля).
Необходимо вывести последовательность C длины $n + m - 1$, определенную следующим образом:

$$C_k = \sum_{i=0}^{\lfloor \frac{k-1}{2} \rfloor} A_i \cdot B_{k-i}$$

Считайте, что если A_i или B_{k-i} не определены, то они равны нулю.

Формат входных данных

На первой строке ввода находится целое число n ($1 \leq n \leq 500000$).

На второй строке даны n целых чисел, являющиеся элементами последовательности A ($0 \leq A_i \leq 100$).

На третьей строке ввода находится целое число m ($1 \leq m \leq 500000$).

На четвертой строке даны m целых чисел, являющиеся элементами последовательности B ($0 \leq B_i \leq 100$).

Формат выходных данных

Выведите $n + m - 1$ чисел, являющихся элементами последовательности C .

Пример

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

Задача I. Особые позиции

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

Будучи школьником Стас очень грустил, когда рассадка в школьном автобусе не позволяла ему сесть рядом с другом, поэтому сейчас он вырос и начал писать программу, решающую эту проблему, и ему потребовалось решить такую задачу.

Дан массив a длины n . Также даны m различных позиций $p_1, p_2, \dots, p_m$ ($1 \leq p_i \leq n$).

Затем равновероятно выбирается **непустое** подмножество этих позиций T и вычисляется

$$\sum_{i=1}^n (a_i \cdot \min_{j \in T} |i - j|).$$

Иными словами, для каждого индекса массива перемножаются a_i и расстояние до ближайшей выбранной в подмножество позиции, и эти величины суммируются.

Найдите математическое ожидание этой величины.

Это число нужно найти по модулю 998 244 353. Формально, пусть $M = 998\,244\,353$. Можно показать, что ответ может быть представлен в виде несократимой дроби $\frac{p}{q}$, где p и q целые числа и $q \not\equiv 0 \pmod{M}$. Выведите целое число, равное $p \cdot q^{-1} \pmod{M}$. Другими словами, выведите такое целое число x , что $0 \leq x < M$ и $x \cdot q = p \pmod{M}$.

Формат входных данных

В первой строке находится два целых числа n и m ($1 \leq m \leq n \leq 10^5$).

Во второй строке находятся n целых чисел $a_1, a_2, \dots, a_n$ ($0 \leq a_i < 998\,244\,353$).

В третьей строке строке находятся m различных целых чисел $p_1, p_2, \dots, p_m$ ($1 \leq p_i \leq n$).

Для всех $1 \leq i < m$ гарантируется, что $p_i < p_{i+1}$.

Формат выходных данных

Выведите единственное целое число — ответ на задачу.

Примеры

стандартный ввод	стандартный вывод
4 2 1 2 3 4 1 4	665496247
6 6 4 2 4 2 4 2 1 2 3 4 5 6	855638030

Замечание

В первом примере:

- Если взята только позиция 1, то итоговая величина равна $1 \cdot 0 + 2 \cdot 1 + 3 \cdot 2 + 4 \cdot 3 = 20$.
- Если взята только позиция 4, то итоговая величина равна $1 \cdot 3 + 2 \cdot 2 + 3 \cdot 1 + 4 \cdot 0 = 10$.
- Если взяты обе позиции, то итоговая величина равна $1 \cdot 0 + 2 \cdot 1 + 3 \cdot 1 + 4 \cdot 0 = 5$.

Ответ на задачу $\frac{20+10+5}{3} = \frac{35}{3} = 665\,496\,247$ (по модулю 998 244 353).

Задача J. Кохиа и скобки

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 3 секунды
Ограничение по памяти: 512 мегабайт

У Чио есть скобочная последовательность[†] s длины n . Пусть k равно минимальному числу символов, которое Чио должна удалить из s , чтобы сделать s правильной[‡].

Теперь Косия хочет, чтобы вы посчитали количество способов удалить k символов из s так, чтобы s стала правильной, по модулю 998 244 353.

Обратите внимание, что два способа удалить символы считаются различными, если и только если множества удаленных индексов различаются.

[†] Скобочной последовательностью называется строка, состоящая только из символов «(» и «)».

[‡] Скобочная последовательность называется правильной, если ее можно превратить в корректное математическое выражение, добавляя только символы $+$ и 1 . Например, последовательности $((()))$, $()$, $((())())$ и пустая строка являются правильными, а $)()$, $((()$ и $((()))()$ — нет.

Формат входных данных

Первая строка содержит строку s ($1 \leq |s| \leq 5 \cdot 10^5$) — скобочную последовательность.

Гарантируется, что s содержит только символы «(» и «)».

Формат выходных данных

Выведите одно целое число — количество способов удалить k символов из s так, чтобы s стала правильной, по модулю 998 244 353.

Примеры

стандартный ввод	стандартный вывод
$((()))$	4
$($	1

Замечание

В первом примере можно показать, что минимальное количество символов, которое нужно удалить, равно 2. Есть 4 способа удалить 2 символа так, чтобы s стала правильной, как показано ниже. Удаленные символы показаны красным.

- $(\textcolor{red}{)})$ $(()$,
- $(())\textcolor{red}{)}$,
- $(\textcolor{red}{()})$,
- $(())\textcolor{red}{(}$.

Во втором примере единственный способ сделать s сбалансированной — удалить единственную скобку, чтобы получить пустую скобочную последовательность, которая считается сбалансированной.