

Задача А. Хорошая матрица

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 6 секунд
Ограничение по памяти: 256 мегабайт

Назовём матрицу, состоящую из нулей и единиц *хорошей*, если в ней нет двух единиц, соседних по стороне.

Назовём матрицу, состоящую из нулей и единиц, *связной*, если между любой парой нулей существует путь, не проходящий по единицам, каждые две клетки которого имеют общую сторону.

Сколько существует *хороших связных* матриц из нулей и единиц, состоящих из n строк и m столбцов? Поскольку ответ может быть слишком большим, посчитайте его остаток по простому модулю p .

Формат входных данных

В единственной строке дано три целых числа n, m, p — количество строк, столбцов в матрице и число, по модулю которого нужно посчитать ответ ($2 \leq n \leq 11$; $1 \leq m \leq 10^9$; $2 \leq p \leq 10^9$, p — простое).

Формат выходных данных

Выведите одно целое число — искомое количество матриц по модулю p .

Примеры

стандартный ввод	стандартный вывод
2 2 998244353	5
4 1 998244353	4
4 5 998244353	2749

Задача В. Определитель

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 3 секунды
Ограничение по памяти: 256 мегабайт

Дана матрица A из n строк и столбцов, в которой мало ненулевых значений. Посчитайте ее определитель по модулю 998 244 353.

Формат входных данных

В первой строке дано два целых числа n и k — размер матрицы и количество ненулевых элементов в ней ($1 \leq n \leq 2500, 1 \leq k \leq 7500$).

В следующих k строках дано по три целых числа x_i, y_i, a_i — ячейка матрицы и значение в ней ($1 \leq x_i, y_i \leq n; 1 \leq a_i < 998\,244\,353$). Все перечисленные ячейки различны.

Формат выходных данных

Выведите одно целое число — значение определителя матрицы по модулю 998 244 353.

Примеры

стандартный ввод	стандартный вывод
2 4 1 1 1 1 2 2 2 1 3 2 2 4	998244351
3 9 1 1 1 1 2 2 1 3 3 2 1 2 2 2 4 2 3 6 3 1 3 3 2 6 3 3 9	0
1 1 1 1 5	5

Задача С. Сумма на спирали

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

Рассмотрим квадратную таблицу $n \times n$ клеток, расположим в ней числа от 1 до n^2 по спирали, начиная с левого верхнего угла по часовой стрелке:

1	2	3	4
12	13	14	5
11	16	15	6
10	9	8	7

Рассмотрим все прямоугольники, которые можно вырезать из этой таблицы. Посчитаем сумму в каждом таком прямоугольнике. Выведите сумму этих сумм по всем прямоугольникам, по модулю $10^9 + 7$.

Формат входных данных

Ввод содержит число n ($1 \leq n \leq 10^6$).

Формат выходных данных

Выведите требуемую сумму по модулю $10^9 + 7$.

Примеры

стандартный ввод	стандартный вывод
1	1
2	40
4	3784
100	201678938
1000000	11304059

Задача D. Подмногоугольник

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

Пусть P — (строго) выпуклый многоугольник, пусть V — множество вершин P , и пусть W — любое подмножество в V размера 3 или более. Тогда выпуклая оболочка множества W назовем подмногоугольником в P .

Например, пусть $(0, 0)$, $(2, 0)$, $(3, 3)$, $(0, 2)$ — вершины выпуклого многоугольника P . Этот многоугольник имеет пять подмногоугольников: сам многоугольник и четыре треугольника.

Вам дан специальный выпуклый многоугольник P из $4n$ вершин, все из которых имеют целочисленные координаты. Вычислите сумму площадей всех подмногоугольников в P .

Процедура генерации входного многоугольника P заключается в следующем. Вам дано целое число n . Рассмотрим квадрат с вершинами $(n, 0)$, $(0, n)$, $(-n, 0)$, $(0, -n)$. На границе этого квадрата ровно $4n$ точек с целочисленными координатами (включая четыре угла квадрата.) Начните с произвольной из этих $4n$ точек и обозначьте их от $T[0]$ до $T[4n - 1]$ в порядке обхода против часовой стрелки.

Пусть $P[0]$ — произвольная точка на плоскости. Для всех i от 1 до $4n$ установим $P[i] = P[i - 1] + T[i - 1]$. То есть, если $P[i - 1]$ имеет координаты (x_1, y_1) , а $T[i - 1]$ имеет координаты (x_2, y_2) , новая точка $P[i]$ имеет координаты $(x_1 + x_2, y_1 + y_2)$.

Полигон P , который вы должны обработать, — это многоугольник с вершинами $P[0], P[1], \dots, P[4n - 1]$. Обратите внимание, что P всегда является простым выпуклым многоугольником, и что ответ, который вы должны вернуть, не зависит от выбора, который вы сделали во время построения. Также отметим, что описанный выше процесс устанавливает $P[4n]$ в ту же точку, что и $P[0]$.

Формат входных данных

Ввод содержит одно число n ($1 \leq n \leq 125\,000$).

Формат выходных данных

Выведите сумму площадей всех подмногоугольников P . Выразите ответ в виде дроби A/B , где $\gcd(A, B) = 1$. Пусть B^{-1} — число, обратное B по модулю 998244353. Выведите число AB^{-1} по модулю 998244353.

Примеры

стандартный ввод	стандартный вывод
1	3
2	1650
35	132946800

Замечание

В первом примере многоугольник P равен $(0, 0)$, $(1, 0)$, $(0, 1)$, $(1, 1)$. У него есть пять подмногоугольников: сам P имеет площадь 1, и у каждого из четырех треугольников площадь $1/2$. Сумма этих площадей равна 3.

Задача Е. Ожерелье

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

Ювелир должен сделать эксклюзивное ожерелье для Королевы. Ожерелье должно состоять из серебряных, золотых и бронзовых бусинок, расположение которых строго специфицировано. Золотые бусинки одинаковые и могут использоваться взаимозаменяемо, аналогично обстоит дело с серебряными и бронзовыми бусинками. Ювелир подготовил бусины для работы и нанизал их на один длинный стержень. Теперь он готов собирать ожерелье снимая бусины одну за одной со стержня и нанизывая на шнурок с любой из сторон, а в завершение процесса соединяя два конца шнурка. Соединение будет незаметно, поэтому оно может быть между любыми двумя бусинами.

К несчастью, бусины на стержне могут не быть в том же порядке, в котором они появятся на ожерелье. Поэтому в процессе сборки ожерелья, ювелир может брать бусины со стержня и откладывать в сторону. Ювелир хочет минимизировать максимальное количество бусин, которые он отложит в сторону в процессе сборки ожерелья.

Формат входных данных

Первая строка ввода содержит одно целое число L ($1 \leq L \leq 1000$) — количество бусин в ожерелье. Следующая строка содержит строку из L букв (каждая из которых либо G , либо S , либо B , означающая золотую, серебряную или бронзовую бусину), которая описывает финальное состояние ожерелья (разрезанного в произвольной точке и выпрямленного). Третья строка содержит строку из L букв, описывающую порядок бусин на палочке. Ювелир может брать бусины только с левого конца палочки. Гарантируется, что возможно собрать ожерелье из заданного расположения бусин.

Формат выходных данных

Вывод должен содержать одну строку — минимально возможное максимальное количество бусин, которые ювелир отложит в сторону в процессе сборки ожерелья.

Примеры

<code>necklace.in</code>	<code>necklace.out</code>
8 GSGSGSGS SSSSGGGG	3
8 SSSGGGBB GSGSGSBB	0

Задача F. Генератор случайных чисел

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 7 секунд
Ограничение по памяти: 256 мегабайт

Одним из возможных способов написать генератор случайных чисел являются линейные рекурренты.

Рассмотрим следующую линейную рекурренту:

$$A_i = (A_{i-1}C_1 + A_{i-2}C_2 + \dots + A_{i-k}C_k) \bmod 104857601, \text{ где } i \geq k + 1$$

Вам даны начальные значения $A_1, A_2, \dots, A_k$, а также коэффициенты рекурренты $C_1, C_2, \dots, C_k$.

Вычислите A_n , для заданного n .

Формат входных данных

В первой строке дано число k ($1 \leq k \leq 30\,000$), и число n ($1 \leq n \leq 10^9$).

Вторая строка содержит ровно k чисел: $A_1, A_2, \dots, A_k$ ($0 \leq A_i < 104857601$).

В третьей строке записаны ровно k чисел: $C_1, C_2, \dots, C_k$ ($0 \leq C_i < 104857601$).

Формат выходных данных

Выведите одно число — ответ на задачу.

Примеры

стандартный ввод	стандартный вывод
3 5 1 2 3 4 5 6	139