Задача A. Операции с многочленами

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

Даны два многочлена P и Q:
$P(t) = p_0 + p_1 \cdot t + \dots + p_n \cdot t^n$
$Q(t) = q_0 + q_1 \cdot t + \dots + q_m \cdot t^m$

Найдите $P(t) + Q(t)$, $P(t) \cdot Q(t)$ и первые 1000 коэффициентов ряда $\frac{P(t)}{Q(t)}$. Все вычисления необходимо производить по модулю $998\,244\,353$.

Формат входных данных

В первой строке содержатся числа n и m ($1 \leqslant n, m \leqslant 1000$) — степени многочленов P и Q.

Вторая строка содержит $n+1$ число $p_0, p_1, \dots, p_n$ — коэффициенты многочлена P ($0 \leqslant p_i < 998\,244\,353$), гарантируется, что $p_n > 0$.

Третья строка содержит $m+1$ число $q_0, q_1, \dots, q_m$ — коэффициенты многочлена Q ($0 \leqslant q_i < 998\,244\,353$), гарантируется, что $q_0 = 1$ и $q_m > 0$.

Формат выходных данных

В первой строке выведите степень многочлена $P + Q$, во второй строке выведите его коэффициенты. Если многочлен не равен тождественно нулю, то старший коэффициент должен быть ненулевым, степень многочлена, тождественно равного нулю, считается равной 0.

В третьей строке выведите степень многочлена $P \cdot Q$, во четвертой строке выведите его коэффициенты, старший коэффициент должен быть ненулевым.

В последней строке выведите 1000 первых коэффициентов $\frac{P(t)}{Q(t)}$.

Примеры

стандартный ввод	стандартный вывод
3 2 0 1 2 3 1 2 3	3 1 3 5 3 5 0 1 4 10 12 9 0 1 0 ... 0
1 3 1 2 1 4 5 2	3 2 6 5 2 4 1 6 13 12 4 1 998244351 3 ... 999 998243353

Задача B. Операции с многочленами — 2

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

Дан многочлен P степени n со нулевым свободным членом:

$$P(t) = p_1 \cdot t + \ldots + p_n \cdot t^n$$

Найдите первые m коэффициентов $\sqrt{1+P(t)}$, $e^{P(t)}$ и $\ln(1+P(t))$. Все вычисления необходимо производить по модулю $998\,244\,353$.

Формат входных данных

В первой строке содержатся числа n и m $(1 \leqslant n, m \leqslant 100)$ — степень многочлена P и необходимое количество коэффициентов.

Вторая строка содержит $n+1$ число $p_0, p_1, \ldots, p_n$ — коэффициенты многочлена P $(0 \leqslant p_i < 998\,244\,353)$, гарантируется, что $p_n > 0$ и $p_0 = 0$.

Формат выходных данных

Выведите три строки. В первой строке выведите первые m коэффициентов ряда $\sqrt{1+P(t)}$, соответствующие степеням t^0, t^1, …, t^{m-1}. В следующих двух строчках в аналогичном формате выведите коэффициенты $e^{P(t)}$ и $\ln(1+P(t))$ по модулю $998\,244\,353$.

Пример

стандартный ввод	стандартный вывод

`1 4`
`0 1`
	`1 499122177 124780544 935854081`
	`1 1 499122177 166374059`
	`0 1 499122176 332748118`

Замечание

Дробь $\frac{a}{b} \bmod m$ следует вычислять, как $a \cdot b^{-1} \bmod m$, где b^{-1} обозначает обратный по модулю m элемент к b: $bb^{-1} \bmod m = 1$.

Например, $\sqrt{1+t} = 1 + \frac{t}{2} - \frac{t^2}{8} + \frac{t^3}{16} + \ldots$. $\frac{1}{2} \bmod M = 1 \cdot 2^{-1} \bmod M = 499122177$ и $\frac{1}{8} = 1 \cdot 6^{-1} \bmod M = 124780544$.

Задача C. Подсчет деревьев

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

Заданы числа $c_1, c_2, \ldots, c_k$. Посчитайте количество различных бинарных деревьев, в которых вершины могут иметь вес c_i. Вершины равного веса считаются одинаковыми.

Формат входных данных

В первой строке содержатся два целых числа k и m ($1 \leqslant k, m \leqslant 2\,000$) — количество весов вершин и максимальный вес дерева. В следующей строке содержатся числа c_i ($1 \leqslant c_i \leqslant m$). Все c_i различны.

Формат выходных данных

Выведите m чисел — количество деревьев веса $1, 2, \ldots, m$ по модулю $10^9 + 7$.

Примеры

стандартный ввод	стандартный вывод
2 5 1 3	1 2 6 18 57
1 10 2	0 1 0 2 0 5 0 14 0 42

Задача D. Конструируемые комбинаторные классы

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

В этой задаче мы используем следующие способы конструирования комбинаторных объектов.

Базовое множество B состоит из одного объекта u с весом 1. Каждый сконструированный объект x имеет некоторый вес $w(x)$. Если объект сконструирован из одного или нескольких других объектов, его вес равен сумме весов этих объектов.

Пусть X задаёт некоторое множество комбинаторных объектов. Рассмотрим следующие способы создать новые множества объектов.

Множество $L(X)$ состоит из всех возможных списков конечной длины, каждый элемент которых имеет положительный вес и принадлежит множеству X. Например, $L(B)$ состоит из списков $[]$, $[u]$, $[u, u]$, $[u, u, u]$, и так далее. Аналогично, $L(L(B))$ состоит из $[]$, $[[u]]$, $[[u], [u]]$, $[[u, u], [u]]$, $[[u], [u, u]]$, и так далее. Обратите внимание, последние два списка различны, поскольку для списка важен порядок элементов в нем. Также обратите внимание, что $[[]]$ не является корректным списком в $L(L(B))$, поскольку только объекты положительного веса разрешаются в качестве элементов списков, а $[]$ имеет вес 0.

Множество $S(X)$ содержит все возможные мультимножества конечного размера, каждый элемент которых имеет положительный вес и принадлежит X. Например, $S(B)$ состоит из мультимножеств $\{\}$, $\{u\}$, $\{u, u\}$, $\{u, u, u\}$, и так далее. Еще один пример: $S(L(B))$ содержит, например, мультимножества $\{[u]\}$, $\{[u], [u]\}$. Обратите внимание, что мультимножество может содержать несколько равных объектов. Заметьте, что в отличие от списков для мультимножеств не важен порядок элементов, поэтому мультимножество $\{[u], [u, u]\}$ совпадает с мультимножеством $\{[u, u], [u]\}$.

Вес списка или мультимножества равен сумме весов его элементов, например, вес $([u], [u, u], [u, u, u])$ равен 6.

Наконец, последний рассматриваемый способ создания нового типа комбинаторных объектов — пара. Если X и Y — множества комбинаторных объектов, то $P(X, Y)$ представляет собой множество упорядоченных пар объектов, где первый компонент взят из X, а второй — из Y. Например, $P(S(B), L(B))$ содержит в качестве элементов $\langle\{u, u\}, [u, u, u]\rangle$ и $\langle\{\}, [u]\rangle$. Обратите внимание, что в отличие от списков, мультимножеств и циклов, пары могут содержать компоненты нулевого веса.

По заданному описанию класса комбинаторных объектов посчитайте количество элементов веса 0, 1, 2, 3, 4, 5 и 6.

Формат входных данных

В единственной строке входного файла содержится корректное описание комбинаторного объекта. Длина описания не превосходит 200.

Формат выходных данных

Выведите семь целых чисел — количество объектов в описанном комбинаторном классе с весом от 0 до 6.

Примеры

стандартный ввод
`P(S(B),L(B))`
стандартный вывод
`1 2 3 4 5 6 7`

стандартный ввод
`S(L(B))`
стандартный вывод
`1 1 2 3 5 7 11`

стандартный ввод
L(P(L(L(L(P(P(P(B,L(B)),L(B)),P(B,L(B)))))),P(B,L(B))))
стандартный вывод
1 1 2 5 14 42 132

Задача E. Деревья, избегающие левых расчёсок

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

Структуры, избегающие определенных подструктур, активно изучаются в комбинаторике. В этой задаче мы изучим деревья, избегающие определенных поддеревьев.

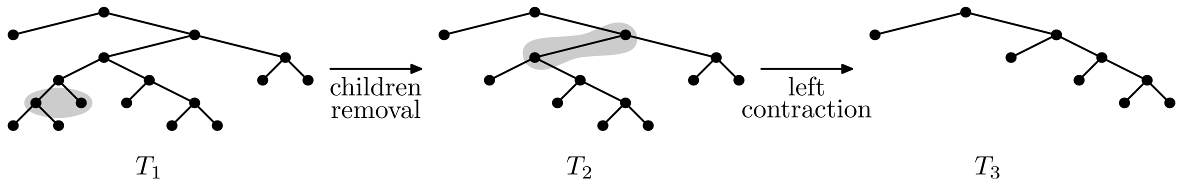
Рассмотрим подвешенное двоичное дерево, в котором каждая вершина имеет ровно двух детей: левого и правого (внутренняя вершина), или не имеет ни одного ребенка (лист). В особом случае дерева из одной вершины его корень также считается листом.

Будем говорить, что дерево T стягивается к дереву R, если R можно получить из T последовательностью следующих операций:

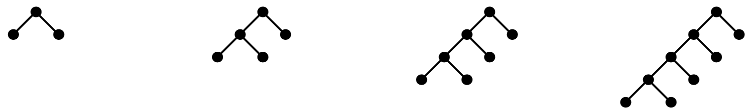
- Удаление детей: удалить оба поддерева у внутренней вершины, превратив ее в лист.
- Левое стягивание: пусть y — левый сын x. Заменим детей x на детей y.
- Правое стягивание: пусть y — правый сын x. Заменим детей x на детей y.

Дерево T избегает дерева R, если T не стягивается к дереву R.

Рисунок ниже показывает описанные операции, также он демонстрирует, что дерево T_1 стягивается к дереву T_3.

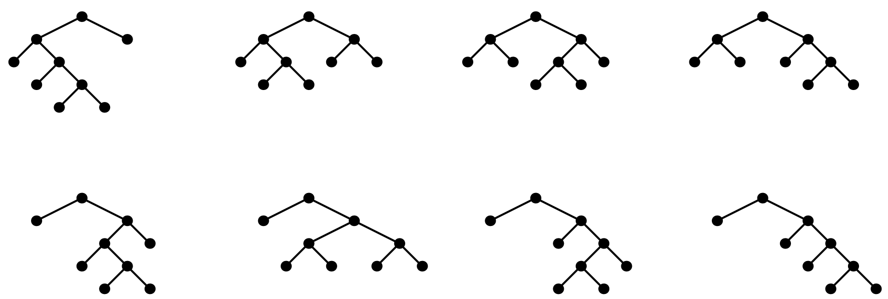


Левой расческой порядка k называется дерево с k листьями, где правый сын любой вершины представляет собой лист. На рисунке ниже показаны левые расчески порядка k для k от 2 до 5.



По заданному k и n вычислите для всех i от 1 до n количество деревьев с i листьями, избегающих левых расчесок порядка k. Выведите эти числа по модулю 998 244 353.

Все деревья с 5 листьями, избегающие левых расчесок порядка 4, показаны на рисунке.



Формат входных данных

На вход подаётся два числа: k и n ($2 \leqslant k \leqslant 5000$, $1 \leqslant n \leqslant 5000$).

Формат выходных данных

Выведите n целых чисел: для каждого i от 1 до n выведите число деревьев с i листьями, избегающих левых расчесок порядка k, выводите числа по модулю 998 244 353.

Примеры

стандартный ввод	стандартный вывод
4 5	1 1 2 4 8
7 6	1 1 2 5 14 42

Задача F. Генератор случайных чисел

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

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

Рассмотрим следующую линейную рекурренту:

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

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

Вычислите A_n, для заданного n.

Формат входных данных

В первой строке дано число k ($1 \leqslant k \leqslant 1000$), и число n ($1 \leqslant n \leqslant 10^{18}$).

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

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

Формат выходных данных

Выведите одно число — ответ на задачу.

Пример

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