

Задача Refrain. Рефрен

Имя входного файла: `refrain.in`
 Имя выходного файла: `refrain.out`
 Ограничение по времени: 3 секунды
 Ограничение по памяти: 64 мегабайта

Рассмотрим последовательность n целых чисел от 1 до m . Подпоследовательность подряд идущих чисел называется *рефреном*, если произведение ее длины на количество вхождений в последовательность максимально.

По заданной последовательности требуется найти ее рефрен.

Формат входного файла

Первая строка входного файла содержит два целых числа: n и m ($1 \leq n \leq 150\,000$, $1 \leq m \leq 10$).

Вторая строка содержит n целых чисел от 1 до m .

Формат выходного файла

Первая строка выходного файла должна содержать произведение длины рефрена на количество ее вхождений. Вторая строка должна содержать длину рефрена. Третья строка должна содержать последовательность которая является рефреном.

Пример

refrain.in	refrain.out
9 3	9
1 2 1 2 1 3 1 2 1	3
	1 2 1

Задача Refrain-Easy. Рефрен (простая)

Задача «Рефрен» с ограничением 100 на количество чисел и 10 на сами числа.

Задача Spell. Заклинание

Имя входного файла: `spell.in`
 Имя выходного файла: `spell.out`
 Ограничение по времени: 2 секунды
 Ограничение по памяти: 64 мегабайта

Троль Вол'Джин ищет наилучшие лечащие заклинания. Сейчас он пытается соединить древние заклинания с современными исследованиями в области лечащей магии, чтобы его заклинания лечили быстрее, чем любые известные. Вол'Джин взял N заклинаний из древних книг. Он хочет, чтобы его новое заклинание было бы частью всех этих древних заклинаний. Это значит, что если произнести любое из древних заклинаний, новое заклинание Вол'Джина тоже будет произнесено. Троль кодирует заклинания строчными буквами латинского алфавита, для того, чтобы их было проще записывать. Конечно же, может существовать несколько различных заклинаний, являющихся частью каждого из древних заклинаний, выбранных Вол'Джином. Поэтому он не просит Вас найти их все, а просит среди всех таких заклинаний вывести некоторые по их номерам в лексикографическом порядке. Вашей задачей будет найти все заклинания, запрошенные Вол'Джином.

Формат входного файла

В первой строке входного файла находится ровно одно число N ($1 \leq N \leq 20$) — количество древних заклинаний, выбранных Вол'Джином. Далее следует N строк с описаниями этих заклинаний. Каждое описание заклинания — строка из маленьких латинских букв. Сумма длин всех описаний не превосходит 10^5 . После этого следует число M — количество заклинаний, которые Вол'Джин хочет получить. Далее следует M строк, по одному числу в каждой — позиция в лексикографическом порядке (пронумерованном с единицы) заклинания, которое Вол'Джин хочет получить. Гарантируется, что

эта позиция не превосходит общего количества подходящих заклинаний.

Формат выходного файла

Для каждого запроса выведите соответствующее заклинание. Каждое заклинание выводите на отдельной строке. Гарантируется, что суммарная длина всех ответов не превосходит 10^5 .

Пример

spell.in	spell.out
2	a
abacaba	ab
dabacaba	aba
16	abac
1	abaca
2	abacab
3	abacaba
4	ac
5	aca
6	acab
7	acaba
8	b
9	ba
10	bac
11	baca
12	bacab
13	
14	
15	
16	

Задача Suffix. Суффиксный пулемёт

Имя входного файла: `suffix.in`
 Имя выходного файла: `suffix.out`
 Ограничение по времени: 2 секунды
 Ограничение по памяти: 64 мегабайта

Или зачёт, или автомат.

Ганнибал Ректор

Теоретическая подготовка новобранцев армии Поссилтума включала в себя не только занятия по военному праву, но и начала криптографии. Лекции читал майор Мега Байт, не чуждый солдатского юмора. Гвидо и Нунцио, в чьё задание входил развал армии Поссилтума изнутри, решили на этом сыграть, внося путаницу в терминологию. В начале очередной лекции Нунцио поднял руку и спросил:

— Вот вы на прошлой лекции рассказывали про конечные автоматы. А про конечные пулемёты расскажете?

Мега Байт не растерялся.

— Суффиксный пулемёт — это конечный автомат, принимающий все суффиксы данной строки (от нулевого до L -го включительно, где L — длина строки), и только их. Сержант Гвидо!

— Я, господин майор!

— Вы сможете отличить автомат от пулемёта?

— Так точно, господин майор!

— Вам дан конечный автомат. Требуется проверить, является ли он суффиксным пулемётом данной строки.

К сожалению, написание программ такого типа не входило в обязанности Гвидо и Нунцио как в Синдикате, так и в корпорации М. И. Ф. Так что соответствующую программу придётся писать Вам.

Формат входного файла

Во входном файле задан один или несколько тестовых наборов. В первой строке каждого набора заданы количество состояний автомата N , количество переходов M , а также количество принимающих состояний T ($1 \leq T \leq N \leq 50\,000$, $1 \leq M \leq 100\,000$). Во второй строке через пробел заданы T различных чисел в пределах от 1 до N — принимающие состояния автомата, в возрастающем порядке. В последующих M строках заданы переходы в виде $a_i \ b_i \ c_i$, где $1 \leq a_i, b_i \leq n$, а c_i — маленькая буква латинского алфавита. Переход производится из состояния a_i в состояние b_i по букве c_i . Из каждого состояния a_i есть не более одного перехода по символу c_i . Последняя строка описания набора — это строка S , для которой автомат должен являться пулемётом. Она состоит только из маленьких латинских букв, и ее длина лежит в пределах от 1 до 50 000 включительно. Кроме того, сумма всех N и суммарная длина всех строк, для которых необходимо произвести проверку, не превосходит 50 000, а сумма всех M не превосходит 100 000.

Файл заканчивается фиктивным набором, в котором $N = M = T = 0$.

Начальным состоянием автомата является первое. Если при интерпретации какой-то строки в автомате отсутствует соответствующий переход, то автомат вываливается по ошибке и строку не принимает. Таким образом, строка принимается, только если при её интерпретации были найдены все переходы, и по их завершении автомат оказался в принимающем состоянии (при этом неважно, были по пути принимающие состояния, или нет).

Формат выходного файла

Выведите в выходной файл, является ли данный автомат пулемётом, следуя формату примера.

Пример

suffix.in
2 1 2
1 2
1 2 a
a
2 2 2
1 2
1 1 a
1 2 b
ab
0 0 0
suffix.out
Automaton 1 is a machinegun.
Automaton 2 is not a machinegun.