Тамотва йынскиффус

Имя входного файла: `notamotua.in`
Имя выходного файла: `notamotua.out`

Дана строка. Ой, простите, не дана. Зато дан суффиксный автомат, по ней постоенный. Но раз строки не было во входном файле, по закону сохранения она должна появиться в выходном. Причем это должна быть та же самая строка!

Формат входного файла

В первой строке входного файла заданы количество состояний автомата N, количество переходов M, а также количество принимающих состояний T ($1 \leqslant T \leqslant N \leqslant 100\,000$, $1 \leqslant M \leqslant 200\,000$). 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 $1 \leqslant a_i, b_i \leqslant n$, а c_i — строчная 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.

Начальным состоянием автомата является первое.

Формат выходного файла

Выведите в выходной файл требуемую строку.

Пример

notamotua.in
11 16 3 1 2 7 1 2 a 1 3 i 1 4 s 1 5 t 1 6 n 2 4 s 2 6 n 3 7 a 4 3 i 4 5 t 5 8 a 6 9 a 8 10 s 9 11 s 10 3 i 11 5 t
notamotua.out
anastasia

Суффиксный пулемёт

Имя входного файла: `suffix.in`
Имя выходного файла: `suffix.out`

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, внеся 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 ($1 \leqslant T \leqslant N \leqslant 50\,000$, $1 \leqslant M \leqslant 100\,000$). 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 $1 \leqslant a_i, b_i \leqslant n$, 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.
```