Задача A. Расстояние

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

В некоторой стране N городов и M дорог. Каждая дорога является двухсторонней и соединяет два различных города, причём известно, что ни на одной дороге нет других городов, кроме тех, что задают её концы. Расстоянием между двумя городами считается минимальная суммарная длина дорог, по которым надо проехать, чтобы попасть из одного города в другой.

Найдите расстояние между двумя заданными городами.

Формат входного файла

В первой строке входного файла заданы два целых числа N и M ($1 \leqslant N \leqslant 10\,000$, $0 \leqslant M \leqslant 100\,000$). Во второй строке записаны два целых числа X и Y — номера городов, между которыми требуется найти расстояние ($1 \leqslant X, Y \leqslant N$). Следующие M строк содержат по три целых числа A_i, B_i и L_i каждая и обозначают, что между городами A_i и B_i есть дорога длины L_i ($1 \leqslant A_i, B_i \leqslant N$, $1 \leqslant L_i \leqslant 10\,000$).

Формат выходного файла

Выведите одно число — расстояние между двумя городами X и Y. Если по заданным дорогам из X в Y добраться невозможно, выведите -1.

Примеры

distance.in	distance.out
2 1 1 2 1 2 3	3
3 2 1 2 1 3 5 2 3 6	11
5 4 1 4 1 2 1 1 3 2 2 3 3 4 5 5	-1
1 0 1 1	0

Задача B. Дорога

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

В Древнем государстве Оссия было два города, между которыми была проложена дорога длиной S метров. Через каждый метр стояли столбики, на каждом из которых по некторому принципу (этот секретный принцип был известен только древним монахам Шамбалы) было написано по букве (а алфавит там у них был латинский). Однажды князь-король Василий I решил, что человек, когда он едет по этой дороге, слишком редко вспоминает о нём. Он решил это исправить. Для этого он повелел на некоторых столбиках вместо буквы написать «Здесь был Вася». По его представлению, человек, проехав любой участок дороги длиной K метров, должен обязательно хоть раз увидеть такую надпись. Иными словами, среди каждых K идущих подряд столбиков должен оказаться хоть один, на котором буква заменена на надпись. При этом, чтобы не слишком раздражать монахов (а они люди обидчивые), Василий I приказал выбрать для надписи такие столбики, чтобы среди стёртых букв оказалось как можно меньше различных букв латинского алфавита.

Помогите боярам выполнить приказ своего повелителя.

Формат входного файла

В первой строке написано одно целое число K ($1 \leqslant K \leqslant 100\,000$). Во второй строке — без пробелов написано S заглавных латинских букв в той последовательности, в которой ими помечены столбики вдоль дороги. Гарантируется, что $K \leqslant S \leqslant 100\,000$.

Формат выходного файла

В первой строке выведите N — минимальное количество различных букв латинского алфавита, которые хотя бы на одном столбике придётся стереть, чтобы написать «Здесь был Вася». В следующих N строках выведите те заглавные буквы латинского алфавита, которые потребуется хоть раз стереть. Буквы можно выводить в любом порядке. Если ответов с минимальным N несколько, можно вывести любой из них.

Примеры

road.in	road.out
2 ABA	1 A
2 ABBAA	2 A B

Задача C. Строки

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

Вася посещает занятия по программированию. К сожалению, недавно он заболел и не смог прийти на лекцию. Он смог узнать лишь, что на лекции проходили поиск подстроки в строке. Вася боится, что на следующей тренировке ему придётся искать подстроку в строке, а он до сих пор не знает, как это делается. Помогите ему!

Формат входного файла

В первых двух строчках входного файла даны две строки $\mathcal{T}$ и $\mathcal{S}$ ($1 \leqslant |\mathcal{T}| \leqslant 1\,000\,000$, $1 \leqslant |\mathcal{S}| \leqslant 1\,000\,000$). Строки состоят только из строчных букв `a-z` латинского алфавита.

Формат выходного файла

Выведите через пробел все начала вхождений строки $\mathcal{S}$ в строку $\mathcal{T}$ в порядке возрастания. Вхождение начинается в позиции k, если $\mathcal{T}_k = \mathcal{S}_1, \mathcal{T}_{k+1} = \mathcal{S}_2, \ldots, \mathcal{T}_{k+|\mathcal{S}|-1} = \mathcal{S}_{|S|}$. Если вхождений нет, выведите единственное слово `none` вместо списка вхождений.

Примеры

strings.in	strings.out
ababcbc abc	3
qwerty asdfgh	none
babaabaaabaaaabaaaaabaaaaaabaaa aaa	7 11 12 16 17 18 22 23 24 25 29
babaabaaabaaaabaaaaabaaaaaabaaa aab	4 8 13 19 26
aa aaaa	none