

Задача А. Наивный поиск подстроки в строке

Имя входного файла: `search1.in`
Имя выходного файла: `search1.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Даны строки p и t . Требуется найти все вхождения строки p в строку t в качестве подстроки.

Формат входного файла

Первая строка входного файла содержит p , вторая — t ($1 \leq |p|, |t| \leq 10^4$). Строки состоят из букв латинского алфавита.

Формат выходного файла

В первой строке выведите количество вхождений строки p в строку t . Во второй строке выведите в возрастающем порядке номера символов строки t , с которых начинаются вхождения p . Символы нумеруются с единицы.

Примеры

search1.in	search1.out
aba	2
abaCaba	1 5

Задача В. Быстрый поиск подстроки в строке

Имя входного файла: search2.in
Имя выходного файла: search2.out
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Даны строки p и t . Требуется найти все вхождения строки p в строку t в качестве подстроки.

Формат входного файла

Первая строка входного файла содержит p , вторая — t ($1 \leq |p|, |t| \leq 10^6$). Строки состоят из букв латинского алфавита.

Формат выходного файла

В первой строке выведите количество вхождений строки p в строку t . Во второй строке выведите в возрастающем порядке номера символов строки t , с которых начинаются вхождения p . Символы нумеруются с единицы.

Примеры

search2.in	search2.out
aba	2
abaCaba	1 5

Задача С. Префикс-функция

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

Постройте префикс-функцию для заданной строки s .

Формат входного файла

Первая строка входного файла содержит s ($1 \leq |s| \leq 10^6$). Строка состоит из букв латинского алфавита.

Формат выходного файла

Выведите значения префикс-функции строки s для всех индексов $1, 2, \dots, |s|$.

Примеры

<code>prefix.in</code>	<code>prefix.out</code>
aaaAAA	0 1 2 0 0 0
abacaba	0 0 1 0 1 2 3

Задача D. Z-функция

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

Постройте Z-функцию для заданной строки s .

Формат входного файла

Первая строка входного файла содержит s ($1 \leq |s| \leq 10^6$). Строка состоит из букв латинского алфавита.

Формат выходного файла

Выведите значения Z-функции строки s для индексов $2, 3, \dots, |s|$.

Примеры

<code>z.in</code>	<code>z.out</code>
aaaAAA	2 1 0 0 0
abacaba	0 1 0 3 0 1

Задача Е. Поиск подстроки в строке с одним несовпадением

Имя входного файла: `search3.in`
Имя выходного файла: `search3.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Даны строки p и t . Требуется найти все вхождения строки p в строку t в качестве подстроки, с точностью до возможного несовпадения одного символа.

Формат входного файла

Первая строка входного файла содержит p , вторая — t ($1 \leq |p|, |t| \leq 10^6$). Строки состоят из букв латинского алфавита.

Формат выходного файла

В первой строке выведите количество вхождений строки p в строку t . Во второй строке выведите в возрастающем порядке номера символов строки t , с которых начинаются вхождения p . Символы нумеруются с единицы.

Примеры

search3.in	search3.out
aaaa	4
Caabdbaaaa	1 2 6 7

Задача F. Поиск периода

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

Дана строка s . Требуется найти минимальную по длине строку t , такую что s представима в виде конкатенации одной или нескольких строк t .

Формат входного файла

Первая строка входного файла содержит s ($1 \leq |s| \leq 10^6$). Строка состоит из букв латинского алфавита.

Формат выходного файла

Выведите длину искомой строки t .

Примеры

<code>period.in</code>	<code>period.out</code>
abcbabcabc	3
abacaba	7

Задача G. Множественный поиск

Имя входного файла: `search4.in`
Имя выходного файла: `search4.out`
Ограничение по времени: 3 секунды
Ограничение по памяти: 256 мегабайт

Дано множество строк S и строка t . Требуется для каждой строки $p \in S$ определить, встречается ли она в t как подстрока.

Формат входного файла

Первая строка входного файла содержит целое число n — мощность S ($1 \leq n \leq 10^6$). Следующие n строк содержат по одной строке из S . Сумма длин всех строк из S не превосходит 10^6 . Последняя строка входного файла содержит t ($1 \leq t \leq 10^6$). Все строки состоят из прописных латинских букв.

Формат выходного файла

Для каждой строки из S выведите «YES», если она встречается в t и «NO» в противном случае. Строки нумеруются в порядке появления во входном файле.

Примеры

search4.in	search4.out
3	YES
abc	NO
abcdr	YES
abcde	
xabcdef	