

## Содержание

<b>Задачи</b>	<b>2</b>
<b>Задача 8A. Помогите, спасите! [1 sec, 256 mb]</b>	<b>2</b>
<b>Задача 8B. LZSS encoding [1 sec, 256 mb]</b>	<b>3</b>
<b>Задача 8C. Набор строк [1 sec, 256 mb]</b>	<b>4</b>
<b>Задача 8D. Ненокку [1 sec, 256 mb]</b>	<b>5</b>

---

В некоторых задачах большой ввод и вывод. Пользуйтесь **быстрым вводом-выводом**.

В некоторых задачах нужен STL, который активно использует динамическую память (set-ы, map-ы) **переопределение стандартного аллокатора** ускорит вашу программу.

## Задачи

### Задача 8А. Помогите, спасите! [1 sec, 256 mb]

Дана строка. Найдите для каждого её префикса количество различных подстрок в нём.

#### Формат входных данных

В единственной строке входных данных содержится непустая строка  $S$ , состоящая из  $N$  ( $1 \leq N \leq 2 \cdot 10^5$ ) маленьких букв английского алфавита.

#### Формат выходных данных

Выведите  $N$  строк, в  $i$ -й строке должно содержаться количество различных подстрок в  $i$ -м префиксе строки  $S$ .

#### Примеры

stdin	stdout
aabab	1 2 5 8 11
atari	1 3 5 9 14

### Задача 8B. LZSS encoding [1 sec, 256 mb]

Алиса хочет отправить сообщение Бобу. Она хочет зашифровать сообщение, используя оригинальный метод шифрования. Сообщение – строка  $S$ , состоящая из  $N$  строчных английских букв.

$S[a \dots b]$  означает подстроку  $S$  от  $S[a]$  до  $S[b]$  ( $0 \leq a \leq b < N$ ). Если первые  $i$  букв уже зашифрованы, Алиса найдёт такие  $(j, k)$ :  $s[j..j+k] = s[i..i+k]$ ,  $k \geq 0$ ,  $0 \leq j < i$ ,  $k = \max$ . Если несколько  $j$  дают максимальное  $k$ , Алиса выберет минимальное  $j$ . Если  $k > 0$  Алиса добавит пару  $\langle j, k \rangle$  в шифр и увеличит  $i$  на  $k$ , иначе Алиса добавит -1 и ASCII код буквы  $S[i]$  в шифр и увеличит  $i$  на 1.

Очевидно шифр начнёт с -1, далее будет ASCII код символа  $S[0]$ . Помогите Алисе реализовать её метод шифрования.

#### Формат входных данных

Первая строка ввода содержит количество тестов  $T$  ( $1 \leq T \leq 50$ ). Следующие  $T$  строк содержат сообщения для шифровки, каждое длины от 1 до  $10^5$ , состоящие из строчных английских букв. Гарантируется, что суммарная длина всех сообщений не превосходит  $2 \cdot 10^6$ .

#### Формат выходных данных

Для каждого теста на отдельной строке выведите “Case #X:”, где  $X$  – номер теста, нумерация с 1. Далее выведите шифр, в каждой строке по два целых числа через пробел.

#### Примеры

stdin	stdout
2	Case #1:
aaaaaa	-1 97
aaaaabbbbbaaabbc	5 0
	Case #2:
	-1 97
	4 0
	-1 98
	4 5
	5 2
	-1 99

### Задача 8С. Набор строк [1 sec, 256 mb]

В Инновационном Отделе НИИ Исследований Данных Строк разработана клавиатура для внутреннего пользования, облегчающая набор строк огромной длины. Кроме обычных клавиш, соответствующих маленьким латинским буквам, на клавиатуре есть еще  $n$  функциональных клавиш  $F_1, \dots, F_n$ , соответствующих заданным строкам из словаря  $S_1, \dots, S_n$ . При нажатии такой клавиши  $F_i$  строка  $S_i$  загружается во внутреннюю память клавиатуры. В каждый момент времени в памяти может находиться не более одной строки из словаря.

Кроме того, в клавиатуру встроен графический манипулятор «Кыш», с помощью которого легким движением руки можно ввести любую подстроку находящейся в памяти строки.

Вася занимается исследованием эффективности данного нововведения. Для этого ему требуется написать программу, которая будет вычислять минимальное необходимое количество действий (нажатий и использований «Кыш») для ввода данной строки  $S$ . В момент начала ввода строки память пуста.

Например, если требуется ввести строку “abacaba”, а в словаре есть строки “baba” и “casa”, то это можно сделать за четыре действия — нажать  $F_1$ , выбрать манипулятором подстроку “aba”, затем нажать ‘с’, и опять выбрать манипулятором подстроку “aba”. Если бы нужно было набрать с таким словарем “bacababa”, то это можно сделать за пять действий: ‘b’,  $F_2$ , “aca”,  $F_1$ , “baba”.

#### Формат входных данных

В первой строке входного файла задано число  $n$  ( $1 \leq n \leq 50$ ). В последующих  $n$  строках заданы  $S_i$ , составленные из не более чем 500 символов. В последней строке вводится непустая строка  $S$ , длина которой не превосходит 100 000. Все символы строк — маленькие латинские буквы.

#### Формат выходных данных

Выведите минимальное необходимое количество действий.

#### Пример

stdin	stdout
2 baba casa abacaba	4
2 baba casa bacababa	5

### Задача 8D. Ненокку [1 sec, 256 mb]

Очень известный автор не менее известной книги решил написать продолжение своего произведения. Он писал все свои книги на компьютере, подключенном к интернету. Из-за такой неосторожности мальчику Ненокку удалось получить доступ к еще ненаписанной книге. Каждый вечер мальчик залазил на компьютер писателя и записывал на свой компьютер новые записи. Ненокку, записав на свой компьютер очередную главу, заинтересовался, а использовал ли хоть раз писатель слово “книга”. Но он не любит читать книги (он лучше ползает в интернете), и поэтому он просит вас узнать есть ли то или иное слово в тексте произведения. Но естественно его интересует не только одно слово, а достаточно много.

#### Формат входных данных

В каждой строчке входного файла записано одна из двух записей.

1. ? <слово> (<слово> — это набор не более 50 латинских символов);
2. A <текст> (<текст> — это набор не более  $10^5$  латинских символов).

1 означает просьбу проверить существование подстроки <слово> в произведение.

2 означает добавление в произведение <текст>.

Писатель только начал работать над произведением, поэтому он не мог написать более  $10^5$  символов. Суммарная длина всех запросов не превосходит 15 мегабайт плюс 12140 байт.

#### Формат выходных данных

Выведите на каждую строчку типа 1 “YES”, если существует подстрока <слово>, и “NO” в противном случае. Не следует различать регистр букв.

#### Пример

stdin	stdout
? love	NO
? is	NO
A Loveis	YES
? love	NO
? WHO	YES
A Whoareyou	
? is	