

Стажировка весна-лето 2023 | бэкенд

7 фев 2023, 16:20:04

старт: 6 фев 2023, 11:48:41

финиш: 6 фев 2023, 16:48:41

длительность: 05:00:00

начало: 1 фев 2023, 00:00:00

Е. Абракадабра

	Все языки	GNU C++20 10.2	Clang14 C++20
Ограничение времени	4 секунды	2 секунды	2 секунды
Ограничение памяти	256Mb	256Mb	256Mb
Ввод	стандартный ввод или input.txt		
Вывод	стандартный вывод или output.txt		

Объявление: если у вас есть **жалобы / претензии / вопросы** насчет задач, то советуем для начала ознакомиться с [расширенной инструкцией](#), содержащей в том числе ответы на самые частые вопросы.

Начало условия: Недавно Кузя разбирал чердак на даче своей бабушки и нашел очень древнюю и непонятную книгу. Кузя сразу понял, что это книга с заклинаниями, а бабушка — волшебница, просто это скрывает.

А раз бабушка умеет творить магию, то и Кузя сможет! Кузя решил тут же применить свои недюжинные навыки чтения и прочитать какие-нибудь заклинания из книги.

В дальнейшем будем считать, что все записи в книге представляют собой одну большую строку S . Все символы в книге представляют собой малые латинские буквы.

Кузя смотрел много фильмов про волшебников, поэтому знает два важных правила:

- Если в данный момент прочитан символ на позиции i , то следующим Кузя должен прочитать символ на позиции p_i ;
- Пусть k_i — порядковый номер в алфавите символа на i -й позиции в тексте (a — 0-й, z — 25-й). Если Кузя **за время одного заклинания** должен прочитать символ на i -й позиции в m_i -й раз, то вместо этого он вслух произносит символ с порядковым номером в алфавите $(k_i + (m_i - 1) \cdot d_i) \bmod 26$.

Подробный пример находится в примечании к тестовым примерам (в самом низу).

Обратите внимание, что изменения символов при прочтении действуют только в рамках одного заклинания (m_i считаются независимо для каждого прочтения заклинания).

Кузя считает, что сила прочитанного заклинания равна количеству уникальных символов, которые в него вошли. К примеру, в заклинании «*zbacbef*» ровно 6 уникальных символов $[a, b, c, e, f, z]$.

Кузя нашел на обложке книги число K и понял, что для оптимального эффекта необходимо прочесть заклинания всех длин от 1 до K включительно, начав по очереди с каждого символа от 1 до N (всего Кузя планирует прочесть $N \times K$ заклинаний).

Кузя боится слишком мощных выбросов магической энергии. Поэтому он просит вас, как победителя викторины по Гарри Поттеру в 5-м классе, заранее вычислить суммарную силу всех $N \times K$ заклинаний, которые он собирается прочесть.

Формат ввода

В первой строке дано два целых числа N и K ($1 \leq N \leq 10^5, 1 \leq K \leq 10^9$) — количество символов в тексте книги.

Во второй строке дана строка S длины N , состоящая из малых латинских букв ($S_i \in [a \dots z]$) — текст книги с заклинаниями.

В третьей строке дано N целых чисел p_i через пробел ($1 \leq p_i \leq N$) — позиция символа, который следует прочесть после чтения символа на i -й позиции.

В четвертой строке дано N целых чисел d_i через пробел ($0 \leq d_i \leq 25$) — сдвиг при повторном чтении символа на i -й позиции (читайте условие и примечание).

Формат вывода

В единственной строке выведите единственное число — суммарную силу всех $N \times K$ заклинаний, начинающихся в каждой из позиций $1, 2, \dots, N$ и имеющих длину $1, 2, \dots, K$.

Пример 1

Ввод	Вывод
3 7 abz 3 1 2 4 0 3	74

Пример 2

Ввод	Вывод
4 6 abcd 2 3 1 4 1 0 2 13	62

Пример 3

Ввод	Вывод
10 1000000000 qwertzxcvb 2 3 4 5 3 4 8 7 10 2 1 2 3 4 5 6 7 8 9 10	259999995297

Примечания

Пояснение к **первому** тестовому примеру.

- Текст в книге равен строке $S = \text{«}abz\text{»}$;
- Массив p равен $[3, 1, 2]$;
- Массив d равен $[4, 0, 3]$;

Разберем детально чтение одного из заклинаний: пусть Кузя начнёт читать заклинание с позиции 3 в тексте и прочтёт 7 символов. В таком случае он прочтёт строку $\text{«}zbacbef\text{»}$:

1. Символ на 3-й позиции равен z . Кузя читает его в $m_3 = 1$ -й раз, поэтому он читает символ z без изменений. После этого Кузя должен прочесть символ на позиции $p_3 = 2$;
2. Аналогично Кузя читает символ b на 2-й позиции в $m_2 = 1$ -й раз, поэтому он читает именно символ b и переходит к символу $p_2 = 1$;
3. Далее Кузя читает символ a на 1-й позиции в $m_1 = 1$ -й раз, поэтому он читает символ a без изменений и переходит к символу $p_1 = 3$;
4. Кузя читает символ на позиции 3, но уже в $m_3 = 2$ -й раз. Из этого следует, что на самом деле Кузя должен произнести не z (25-й в алфавите), а $(25 + (2 - 1) \cdot 3) \bmod 26 = 2$ -й символ в алфавите - это символ c . После чего Кузя снова должен перейти к символу на позиции $p_3 = 2$;
5. Так как $d_2 = 0$, то на 2-й позиции Кузя будет всегда читать один и тот же символ b ;
6. А вот вместо символа a (0-й в алфавите) на позиции 1 Кузя в $m_1 = 2$ -й раз прочтёт символ $(0 + (2 - 1) \cdot 4) \bmod 26 = 4$ -й символ в алфавите — символ e ;
7. Завершает Кузя чтением символа на позиции 3 в $m_3 = 3$ -й раз, поэтому в этот раз он прочтёт $(25 + (3 - 1) \cdot 3) \bmod 26 = 5$ -й символ в алфавите - это символ f .

Список всех заклинаний, которые Кузя прочтёт:

Начиная с позиции 1:

1. a — сила 1;
2. az — сила 2;

3. azb — сила 3;
4. $azbe$ — сила 4;
5. $azbec$ — сила 5;
6. $azbecb$ — сила 5 (символ b уже встречался ранее, поэтому не увеличивает силу);
7. $azbecbi$ — сила 6 (i получился как 4-й символ после e).

Начиная с позиции 2:

1. b — сила 1;
2. ba — сила 2;
3. baz — сила 3;
4. $bazb$ — сила 3;
5. $bazbe$ — сила 4;
6. $bazbec$ — сила 5;
7. $bazbecb$ — сила 5.

Начиная с позиции 3 (подробно описаны выше):

1. z — сила 1;
2. zb — сила 2;
3. zba — сила 3;
4. $zbac$ — сила 4;
5. $zbacb$ — сила 4;
6. $zbacbe$ — сила 5;
7. $zbacbef$ — сила 6.

Суммарная сила всех прочтённых заклинаний равна 74.

Пояснение ко **второму** тестовому примеру.

Список всех заклинаний, которые Кузя прочтёт:

Начиная с позиции 1:

1. a — сила 1;
2. ab — сила 2;
3. abc — сила 3;
4. $abcb$ — сила 3 (b получился как 1-й символа после a);
5. $abcbb$ — сила 3;
6. $abcbbe$ — сила 4 (e получился как 2-й символ после c).

Начиная с позиции 2:

1. b — сила 1;
2. bc — сила 2;
3. bca — сила 3;
4. $bcab$ — сила 3;
5. $bcabe$ — сила 3;
6. $bcabeb$ — сила 4.

Начиная с позиции 3:

1. c — сила 1;
2. ca — сила 2;
3. cab — сила 3;
4. $cabe$ — сила 4;
5. $cabeb$ — сила 4;
6. $cabebb$ — сила 4.

Начиная с позиции 4:

1. d — сила 1;
2. dq — сила 2 (q получился как 13-й символ после d);
3. dqd — сила 2 (d получился как 13-й символ после q);
4. $dqdq$ — сила 2;
5. $dqdqd$ — сила 2;
6. $dqdq dq$ — сила 2.

Суммарная сила всех прочтённых заклинаний равна 62.

Пояснение к **третьему** тестовому примеру.

В данном тесте мы ограничимся только одним из 10 миллиардов прочтённых Кузей заклинаний.

Если начать с позиции 1 и прочесть 7 символов, то получится заклинание «*qwerthv*»:

1. *q* на позиции 1;
2. *w* на позиции 2;
3. *e* на позиции 3;
4. *r* на позиции 4;
5. *t* на позиции 5;
6. *h* на позиции 3 (во 2-й раз вместо 4-го символа в алфавите «*e*» Кузя произнесёт $4 + 3 \cdot (2 - 1) = 7$ -й символ);
7. *v* на позиции 4 (во 2-й раз вместо 17-го символа в алфавите «*r*» Кузя произнесёт $17 + 4 \cdot (2 - 1) = 21$ -й символ).

Язык

Golang 1.16

Набрать здесь

Отправить файл

```
1 package main
2 import (
3     "fmt"
4 )
5
6 func main() {
7     var n, k int
8     var s, alph string
9     var p, d, m []int
10    alph = "abcdefghijklmnopqrstuvwxyz"
11    fmt.Scan(&n, &k)
12    fmt.Scan(&s)
13
14    var t int
15    for i := 0; i < n; i++ {
16        fmt.Scan(&t)
17        p = append(p, t)
18    }
19    for i := 0; i < n; i++ {
20        fmt.Scan(&t)
21        d = append(d, t)
22    }
23    for i := 0; i < n; i++ {
24        m = append(m, 0)
25    }
26
27    var value, zxc int
28    var temp byte
29    var resstr string
30    for i := 0; i < n; i++ {
31        for j := 0; j < n; j++{
32            m[j] = 0
33        }
34        resstr = ""
35        temp = s[i]
36        for j := 0; j < k; j++{
37            if (m[i] == 0){
38                m[i]++
```

Отправить

Предыдущая