

I split U in 3 (abc)

Влад щойно знайшов T рядків U_0, \dots, U_{T-1} , i -й із яких складається з N_i латинських літер у нижньому регістрі.

Оскільки Влад дуже допитливий, йому стало цікаво, для кожного $0 \leq i < T$, скількома способами можна розділити U_i на 3 (можливо порожні) рядки A, B, C так, щоб $U_i = A + B + C$ у кожному з таких сценаріїв:

- Сценарій abc: розбиття повинні задовольняти $A \preceq B \preceq C$;
- Сценарій acb: розбиття повинні задовольняти $A \preceq C \preceq B$;
- Сценарій bac: розбиття повинні задовольняти $B \preceq A \preceq C$;
- Сценарій bca: розбиття повинні задовольняти $B \preceq C \preceq A$;
- Сценарій cab: розбиття повинні задовольняти $C \preceq A \preceq B$;
- Сценарій cba: розбиття повинні задовольняти $C \preceq B \preceq A$.

де $+$ позначає конкатенацію рядків, а \preceq — лексикографічне відношення «не більше».¹

Реалізація

Ви повинні надіслати єдиний файл із розширенням `.cpp`.

← Серед вкладень до цього завдання Ви знайдете шаблон `abc.cpp` із прикладом реалізації.



Один вхідний файл може містити декілька тестів! Переконайтеся, що Ви скидаєте глобальні змінні між різними запусками.

Ви повинні реалізувати таку функцію:

C++

```
void split(int N, string U,
           long long &abc, long long &acb, long long &bac,
           long long &bca, long long &cab, long long &cba);
```

- Ціле число N — довжина рядка U .
- Рядок U — один із рядків, які знайшов Влад.
- Функція має підрахувати відповідь для кожного сценарію, присвоївши значення відповідному аргументу.
- Цю функцію викликають T разів під час виконання Вашої програми.

Грейдер викликатиме функції та друкуватиме повернені значення у вихідний файл.

¹Формально, для двох рядків S і T маємо $S \preceq T$ тоді й лише тоді, коли виконується одне з такого:

- S — порожній рядок;
- Обидва рядки непорожні, і перший символ S стоїть перед першим символом T у латинському алфавіті.
- Обидва рядки непорожні, перші символи однакові, і $S' \preceq T'$, де S' та T' — рядки, отримані видаленням першого символу з S та T відповідно.

Приклад грейдера

Спрощена версія грейдера, доступна в директорії цього завдання. Ви можете використати її, щоб протестувати свої розв'язки локально. Грейдер читає вхідні дані з `stdin`, викликає функцію, яку Ви повинні реалізувати, та записує до `stdout` у такому форматі.

Вхідний файл складається з $T + 1$ рядків, де T — кількість тестів, і містить:

- Рядок 1: ціле число T .
- Рядок $2 + i$ ($0 \leq i < T$): рядок U_i .

Вихідний файл складається з T рядків і містить:

- Рядок $1 + i$ ($0 \leq i < T$): 6 відповідей Вашої програми для i -го тесту у тому ж порядку, у якому вони наведені в умові.

Обмеження

- Загальна довжина рядків в одному вхідному випадку не перевищує 400 000.
- Кожен рядок є непорожнім і складається з латинських літер у нижньому регістрі.

Оцінювання

Вашу програму буде протестовано на кількох тестах, згрупованих у підзадачі. Бал за підзадачу дорівнює найгіршому балу серед її тестів, помноженому на вартість підзадачі.

Бал за тест залежить від кількості правильно розв'язаних шести сценаріїв згідно з такою таблицею:

Розв'язані сценарії	0	1	2	3	4	5	6
Бали	0	0.3	0.5	0.7	0.8	0.9	1

- Підзадача 0 [0 балів]:** Приклади.
- Підзадача 1 [10 балів]:** Єдиний символ у рядку — `a`.
- Підзадача 2 [10 балів]:** Загальна довжина рядків в одному вхідному випадку не перевищує 300.
- Підзадача 3 [20 балів]:** Загальна довжина рядків в одному вхідному випадку не перевищує 15 000.
- Підзадача 4 [60 балів]:** Без додаткових обмежень.

Приклади вводу/виводу

stdin	stdout
3	4 2 5 2 3 2
cafj	8 8 8 8 8 8
aaaaaaa	21 10 9 1 8 1
aabyuxll	

Пояснення

У першому тесті розбиття такі:

- | | | | | | |
|-----------------------|-----------------------|-----------------------|---------------------------|----------------------|---------------------|
| 1. $A = ""$; | $B = ""$; | $C = \text{"cafj"}$. | 9. $A = \text{"c"}$; | $B = \text{"afj"}$; | $C = ""$. |
| 2. $A = ""$; | $B = \text{"c"}$; | $C = \text{"afj"}$. | 10. $A = \text{"ca"}$; | $B = ""$; | $C = \text{"fj"}$. |
| 3. $A = ""$; | $B = \text{"ca"}$; | $C = \text{"fj"}$. | 11. $A = \text{"ca"}$; | $B = \text{"f"}$; | $C = \text{"j"}$. |
| 4. $A = ""$; | $B = \text{"caf"}$; | $C = \text{"j"}$. | 12. $A = \text{"ca"}$; | $B = \text{"fj"}$; | $C = ""$. |
| 5. $A = ""$; | $B = \text{"cafj"}$; | $C = ""$. | 13. $A = \text{"caf"}$; | $B = ""$; | $C = \text{"j"}$. |
| 6. $A = \text{"c"}$; | $B = ""$; | $C = \text{"afj"}$. | 14. $A = \text{"caf"}$; | $B = \text{"j"}$; | $C = ""$. |
| 7. $A = \text{"c"}$; | $B = \text{"a"}$; | $C = \text{"fj"}$. | 15. $A = \text{"cafj"}$; | $B = ""$; | $C = ""$. |
| 8. $A = \text{"c"}$; | $B = \text{"af"}$; | $C = \text{"j"}$. | | | |

Серед них такі підходять для кожного сценарію:

- Сценарій abc: розбиття 1, 3, 4 і 11.
- Сценарій асb: розбиття 2 і 5.
- Сценарій бас: розбиття 1, 7, 8, 10 і 13.
- Сценарій bca: розбиття 6 і 15.
- Сценарій sab: розбиття 5, 12 і 14.
- Сценарій cba: розбиття 9 і 15.