

## I split U in 3 (abc)

Valerio,  $T$  adet  $U_0, \dots, U_{T-1}$  dizisi bulmuştur; bunların  $i$ -incisi  $N_i$  tane Latin küçük harften oluşuyor.

Valerio çok meraklı olduğu için, her  $0 \leq i < T$  için,  $U_i$  dizisini  $U_i = A + B + C$  olacak şekilde 3 adet (boş olabilecek)  $A, B, C$  dizisine kaç farklı şekilde bölebileceğimizi soruyor. Bu bölme işlemleri, aşağıdaki senaryoların her birinin kısıtlamalarına uymalı:

- Senaryo **abc**: bölmeler  $A \preceq B \preceq C$  koşulunu sağlamalı;
- Senaryo **acb**: bölmeler  $A \preceq C \preceq B$  koşulunu sağlamalı;
- Senaryo **bac**: bölmeler  $B \preceq A \preceq C$  koşulunu sağlamalı;
- Senaryo **bca**: bölmeler  $B \preceq C \preceq A$  koşulunu sağlamalı;
- Senaryo **cab**: bölmeler  $C \preceq A \preceq B$  koşulunu sağlamalı;
- Senaryo **cba**: bölmeler  $C \preceq B \preceq A$  koşulunu sağlamalı.

Burada  $+$  string birleştirmeyi,  $\preceq$  ise sözlükbilimsel (yani sözlük sırasına göre) olarak küçük veya eşit olmayı ifade eder.<sup>1</sup>

## Implementasyon

.cpp uzantılı tek bir dosya göndermelisin.

↩ Bu görev için ekler arasında, örnek bir implementasyon içeren **abc.cpp** şablonu bulacaksın.



Tek bir girdi dosyası birden fazla test durumu içerebilir! Farklı çalıştırmalar arasında genel değişkenleri sıfırladığından emin ol.

Aşağıdaki fonksiyonu kodlamalısın:

|     |   |
|-----|---|
| C++ | <pre>void split(int N, string U,<br/>           long long &amp;abc, long long &amp;acb, long long &amp;bac,<br/>           long long &amp;bca, long long &amp;cab, long long &amp;cba);</pre> |
|-----|---|

- $N$  tamsayısı  $U$  dizisinin uzunluğunu temsil eder.
- $U$  dizisi, Valerio'nun bulduğu dizgilerden biridir.
- Fonksiyon, her senaryoyu ilgili parametreye değer atayarak cevaplamalıdır.
- Bu fonksiyon, programının çalışması sırasında  $T$  kez çağrılır.

Değerlendirici (grader) fonksiyonları çağırarak ve dönen değerleri çıktı dosyasına yazacaktır.

## Örnek Değerlendirici

Düzeltilme sırasında kullanılan değerlendiricinin (grader) basitleştirilmiş bir versiyonu, bu problemle ilgili klasörde mevcuttur. Çözümlerini yerel olarak test etmek için kullanabilirsin. Örnek değerlendirici kodlaman gereken fonksiyonu çağırır ve **stdout** dosyasına aşağıdaki formatta yazar.

<sup>1</sup>Resmi olarak, verilen iki  $S$  ve  $T$  dizisi için,  $S \preceq T$  ancak ve ancak aşağıdakilerden biri doğru ise geçerlidir:

- $S$  boş dizidir;
- İki dize de boş değildir ve  $S$ 'nin ilk karakteri Latin alfabesinde  $T$ 'nin ilk karakterinden önce gelir.
- İki dize de boş değildir, iki dizinin ilk karakterleri aynıdır ve  $S'$  ve  $T'$  sırasıyla  $S$  ve  $T$ 'den ilk karakter çıkarılarak elde edilen dizeler olmak üzere  $S' \preceq T'$  geçerlidir.

Girdi dosyası,  $T$  test durumu sayısı olmak üzere  $T + 1$  satırdan oluşur ve şunları içerir:

- Satır 1: bir tamsayı  $T$ .
- Satır  $2 + i$  ( $0 \leq i < T$ ): bir  $U_i$  dizgisi.

Çıktı dosyası,  $T$  satırdan oluşur ve şunları içerir:

- Satır  $1 + i$  ( $0 \leq i < T$ ): programının  $i$ -inci test durumu için verdiği 6 cevap, soru tanımında sunulduğu sırayla.

## Kısıtlamalar

- Bir girdideki dizgilerin toplam uzunluğu en fazla 400 000 olabilir.
- Her bir dizgi boş değildir ve küçük Latin karakterlerden oluşur.

## Puanlama

Programın, alt görevlere ayrılmış çeşitli test durumlarında test edilecektir. Bir alt görev için puan, test durumlarından birinde elde edilen en kötü puanla, alt görevin değeri çarpılarak hesaplanır.

Bir test durumu için puan, aşağıdaki tabloya göre altı senaryodan kaç tanesini doğru çözdüğüne bağlıdır:

| Çözülen senaryolar | 0 | 1   | 2   | 3   | 4   | 5   | 6 |
|--------------------|---|-----|-----|-----|-----|-----|---|
| Puanlar            | 0 | 0.3 | 0.5 | 0.7 | 0.8 | 0.9 | 1 |

- **Alt görev 0 [ 0 puan]:** Örnek durumlar.
- **Alt görev 1 [10 puan]:** Dizgideki tek karakter **a**'dır.
- **Alt görev 2 [10 puan]:** Bir girdideki dizgilerin toplam uzunluğu en fazla 300 olabilir.
- **Alt görev 3 [20 puan]:** Bir girdideki dizgilerin toplam uzunluğu en fazla 15 000 olabilir.
- **Alt görev 4 [60 puan]:** Ek kısıtlama yok.

## Örnekler

| stdin    | stdout        |
|----------|---------------|
| 3        | 4 2 5 2 3 2   |
| cafj     | 8 8 8 8 8 8   |
| aaaaaaa  | 21 10 9 1 8 1 |
| aabyyx11 |               |

## Açıklama

İlk test durumunda bölmeler şunlardır:

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

Bunlardan her senaryo için sayılanlar şunlardır:

- |  |  |
|--|--|
| • Senaryo abc: 1, 3, 4 ve 11 numaralı bölme.     | • Senaryo bca: 6 ve 15 numaralı bölme.     |
| • Senaryo acb: 2 ve 5 numaralı bölme.            | • Senaryo cab: 5, 12 ve 14 numaralı bölme. |
| • Senaryo bac: 1, 7, 8, 10 ve 13 numaralı bölme. | • Senaryo cba: 9 ve 15 numaralı bölme.     |