

I split U in 3 (abc)

Stofl, un génie de l'informatique du village d'à côté, a un objectif un peu fou : dépasser numériquement le Liechtenstein. Il est persuadé que s'il arrive à résoudre des problèmes algorithmiques suffisamment complexes, il pourra, d'une manière ou d'une autre, prendre le contrôle de la principauté. Sa dernière obsession ? Un défi tordu impliquant des chaînes de caractères. Aide-le dans sa quête de domination mondiale (ou du moins, locale) !

Stofl vient de trouver T chaînes de caractères U_0, \dots, U_{T-1} , dont la i -ème est composée de N_i lettres minuscules de l'alphabet latin.

Puisque Stofl est très curieux, il te demande, pour chaque $0 \leq i < T$, de combien de manières on peut découper U_i en 3 chaînes A, B, C (possiblement vides) telles que $U_i = A + B + C$ sous les contraintes de chacun des scénarios suivants :

- Scénario **abc** : les découpages doivent satisfaire $A \preceq B \preceq C$;
- Scénario **acb** : les découpages doivent satisfaire $A \preceq C \preceq B$;
- Scénario **bac** : les découpages doivent satisfaire $B \preceq A \preceq C$;
- Scénario **bca** : les découpages doivent satisfaire $B \preceq C \preceq A$;
- Scénario **cab** : les découpages doivent satisfaire $C \preceq A \preceq B$;
- Scénario **cba** : les découpages doivent satisfaire $C \preceq B \preceq A$.

où $+$ dénote la concaténation de chaînes et \preceq est la relation « inférieur ou égal » selon l'ordre lexicographique.¹

Implémentation

Tu dois soumettre un unique fichier avec l'extension `.cpp`.



Parmi les fichiers attachés à ce problème, tu trouveras un modèle `abc.cpp` avec un exemple d'implémentation.



Un seul fichier d'entrée peut contenir plusieurs cas de test ! Assure-toi de réinitialiser les variables globales entre les exécutions.

Tu dois implémenter la fonction suivante :

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);
```

- L'entier N représente la longueur de la chaîne U .
- La chaîne U est l'une des chaînes que Stofl a trouvées.

¹Formellement, étant donné deux chaînes S et T , on a $S \preceq T$ si et seulement si l'une des conditions suivantes est vraie :

- S est la chaîne vide ;
- Aucune des deux chaînes n'est vide, et le premier caractère de S vient avant le premier caractère de T dans l'alphabet latin.
- Aucune des deux chaînes n'est vide, les premiers caractères des deux chaînes sont identiques et $S' \preceq T'$ où S' et T' sont les chaînes obtenues en enlevant le premier caractère de S et T respectivement.

- La fonction doit donner la réponse pour chaque scénario en assignant les valeurs aux paramètres correspondants.
- Cette fonction est appelée T fois durant l'exécution de ton programme.

Le correcteur appellera les fonctions et écrira les valeurs retournées dans le fichier de sortie.

Évaluateur

Une version simplifiée du correcteur utilisé lors de l'évaluation est disponible dans le dossier de ce problème. Tu peux l'utiliser pour tester tes solutions localement. Le correcteur d'exemple lit les données d'entrée depuis `stdin`, appelle la fonction que tu dois implémenter, et écrit la sortie dans `stdout` au format suivant.

Le fichier d'entrée est composé de $T + 1$ lignes, où T est le nombre de cas de test, et contient :

- Ligne 1 : un entier T .
- Ligne $2 + i$ ($0 \leq i < T$) : une chaîne de caractères U_i .

Le fichier de sortie est composé de T lignes, et contient :

- Ligne $1 + i$ ($0 \leq i < T$) : les 6 réponses que ton programme a données pour le i -ème cas de test, dans le même ordre que celui de l'énoncé.

Contraintes

- La longueur totale des chaînes de caractères dans un fichier d'entrée est au plus 400 000.
- Chacune des chaînes est non vide et est composée de lettres minuscules de l'alphabet latin.

Score

Ton programme sera testé sur plusieurs cas de test regroupés en sous-tâches. Le score pour une sous-tâche est égal au pire score obtenu sur l'un de ses cas de test, multiplié par la valeur de la sous-tâche.

Le score pour un cas de test dépend du nombre de scénarios que tu résous correctement, selon le tableau suivant :

Scénarios résolus	0	1	2	3	4	5	6
Points	0	0.3	0.5	0.7	0.8	0.9	1

- **Sous-tâche 0 [0 points]**: Cas d'exemples.
- **Sous-tâche 1 [10 points]**: Le seul caractère dans la chaîne est `a`.
- **Sous-tâche 2 [10 points]**: La longueur totale des chaînes de caractères dans un fichier d'entrée est au plus 300.
- **Sous-tâche 3 [20 points]**: La longueur totale des chaînes de caractères dans un fichier d'entrée est au plus 15 000.
- **Sous-tâche 4 [60 points]**: Aucune contrainte additionnelle.

Exemples

stdin	stdout
3	4 2 5 2 3 2
cafj	8 8 8 8 8 8
aaaaaa	21 10 9 1 8 1
aabyxll	

Explication

Dans le **premier cas de test**, les découpages sont :

1. $A = ""$; $B = ""$; $C = \text{"cafj"}$.
2. $A = ""$; $B = \text{"c"}$; $C = \text{"afj"}$.

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

Parmi ceux-ci, les suivants comptent pour chaque scénario :

- | | |
|---|---|
| • Scénario abc : découpages 1, 3, 4 et 11. | • Scénario bca : découpages 6 et 15. |
| • Scénario acb : découpages 2 et 5. | • Scénario cab : découpages 5, 12 et 14. |
| • Scénario bac : découpages 1, 7, 8, 10 et 13. | • Scénario cba : découpages 9 et 15. |