

## Tâche 2.2 – Annihilation (annihilation) (100 pts)

En physique des particules, chaque particule a une antiparticule qui lui correspond. Par exemple, l'antiparticule d'un électron est un positron, et inversement. Quand une particule et son antiparticule se rencontrent, les deux disparaissent, et une certaine quantité d'énergie est libérée,<sup>1</sup> phénomène qu'on appelle *annihilation*. Étant en possession d'une large collection de particules et antiparticules, la firme *belgian energy Creation and Production* (beCP) souhaiterait l'utiliser pour générer de l'électricité (ce qui n'est pas vraiment une idée de génie étant donné le nombre de particules nécessaires). Afin de rester profitable, ils peuvent laisser au plus  $K$  particules inutilisées parmi leur collection.

Le production se passe de la manière suivante.  $N$  particules et antiparticules sont initialement placées en une rangée dans de petits compartiments isolés. Les parois de séparations sont ensuite ôtées et l'énergie résultant de l'annihilation de particules adjacentes est capturée. Lorsque des particules s'annihilent, cela donne la chance aux particules qui les entourent de s'annihiler entre elles également, et ainsi de suite. Ce processus doit se passer dans un environnement hautement protégé et un vide absolu.

Pendant l'exécution de cette procédure, les choses peuvent mal se passer : il se peut qu'à un moment les particules adjacentes ne puissent plus s'annihiler entre elles. Une intervention manuelle est naturellement interdite par les régulations de sécurité, mais heureusement la firme a en sa possession un canon à ions. Ils peuvent l'utiliser pour éliminer de la rangée exactement un compartiment par tir, de telle sorte que les compartiments l'entourant deviennent adjacents. Ils désirent bien sûr éviter les pertes, et veulent donc que le nombre d'utilisations  $S$  du canon soit le plus petit possible. Si ce  $S$  minimal devait s'avérer strictement plus grand que  $K$ , toute la procédure devrait alors être annulée et recommencée.

### Tâche

Écrivez un programme qui, étant donné l'ordre sur la rangée des  $N$  particules disponibles et la valeur de  $K$ , calcule le nombre  $S$  si la procédure peut être complétée après utilisation éventuelle du canon, et autrement renvoie  $-1$ , dans le cas où la procédure doit être annulée.

### Limites et contraintes

- $1 \leq N \leq N_{MAX}$ , le nombre de particules et antiparticules ;
- $0 \leq K \leq K_{MAX}$ , le nombre maximal de particules ou antiparticules qui peuvent être éliminées.

	$N_{MAX}$	$K_{MAX}$
Sous-tâche A (7 pts)	400	1
Sous-tâche B (35 pts)	1 000	1
Sous-tâche C (42 pts)	1 000 000	1
Sous-tâche D (16 pts)	400	400

Durée maximale d'exécution : **1 seconde**. Limite mémoire : **64 MB**.

**Attention** : Comme vous pouvez le voir, les sous-tâches sont assez diverses. Toutefois, votre score sera déterminé par votre soumission de score maximal, et non la somme des points de toutes les sous-tâches que vous avez résolues sur la durée du concours. Dès lors, il peut être nécessaire de tester les contraintes ci-dessus sur l'input et d'adapter le code que votre programme exécute en conséquence.

### Input

L'entrée donnée à votre programme aura le format suivant :

- un nombre  $K$  seul sur une ligne, le nombre maximal de particules qui peuvent être éliminées ;
- une unique ligne avec une chaîne de  $N$  particules.

Chaque particule est représentée par une unique lettre de l'alphabet. Une particule  $a$  est une antiparticule de  $b$  si et seulement si  $a$  et  $b$  sont la même lettre, avec  $a$  en majuscule et  $b$  en minuscule ou réciproquement.

X et x sont donc antiparticules l'une de l'autre.

1. ... donnée par la célèbre formule d'Einstein :  $E = mc^2$

**Output**

Votre programme doit écrire sur une seule ligne la valeur minimale de  $S$ , ou  $-1$  si la procédure doit être arrêtée.

**Exemple 1**

Input :

```
1
aABb
```

Output :

```
0
```

Ici s'annihilent simultanément le 'a' avec le 'A' et le 'B' avec le 'b'. Il n'est donc pas nécessaire d'utiliser le canon à ions.

**Exemple 2**

Input :

```
1
BaAab
```

Output :

```
1
```

Si on enlève un des 'a's, le 'A' peut s'annihiler avec le 'a' restant, et après le 'B' peut faire de même avec le 'b' (parce qu'ils sont maintenant l'un à côté de l'autre). Le nombre minimal de tirs est donc égal à 1.

**Exemple 3**

Input :

```
1
beCP
```

Output :

```
-1
```

Aucune particule ne peut s'annihiler avec une autre, donc on devrait dépenser ici 4 tirs. Puisqu'il est seulement permis de tirer une fois, il faut arrêter la procédure.