

### Contraintes

-----

Les solutions au DS sont validées via la plateforme domjudge. Chaque élève a son propre login et son propre mot de passe. Vous pouvez envoyer plusieurs fois une solution pour un même problème sans aucune pénalité.

Vous avez le droit aux supports des cours, TD, TP et aussi à des ressources Internet, mais vous n'avez pas le droit de communiquer avec d'autres personnes. La DSI sauvegarde tout le trafic TCP/IP pendant le DS. Vous pouvez utiliser seulement le protocole http pour accéder aux différents sites Web. Une connexion à un outil de messagerie, réseau social etc invalide automatiquement votre participation au DS avec les conséquences prévues par le règlement des études.

Vous pouvez utiliser le langage C ou le C++, mais sans utiliser la STL.

### Problème numéro 2 : les mots les plus fréquents

-----

Vous disposez en entrée d'un texte formé des caractères 'A', 'B', 'C', 'D' et du séparateur '-'. Dans notre contexte, chaque mot aura exactement 3 caractères. La fréquence d'un mot est définie comme le nombre d'apparitions de ce mot dans le texte. Trouvez le mot le plus fréquent (il peut y avoir plusieurs mots dans cette situation, dans ce cas tous ces mots forment la solution).

Description de l'entrée (IN) : le texte, fini par une fin de ligne (CR + LF). Le texte sera correctement formé de caractères valides, avec des mots de 3 caractères, et il aura la taille maximale de 2048 caractères. Le texte contiendra au moins un mot.

Description de la sortie (OUT) : la fréquence maximale sur la première ligne, suivie des mots les plus fréquents, un par ligne. Les mots seront affichés en ordre lexicographique (A avant B avant C avant D). Observation : chaque ligne affichée sera terminée par deux caractères (CR et LF), correspondant à la chaîne {"\r\n"}.

#### Exemple 1

IN:

ABA-BAB-AAA-ABA-ABA-AAA-CDC

OUT:

3

ABA

#### Exemple 2

IN:

OUT:

0

#### Exemple 3

IN:

DBA-BDA-DDD-BDA-DBA-DBA-AAA-BDA-AAA-AAA-BAA

OUT:

3

AAA

BDA

DBA

## Exemple de code pour les entrées/sorties

```
#include <iostream>
using namespace std;

#define MAX_TEXT_LENGTH 2048+1

int main()
{
    char text[MAX_TEXT_LENGTH];
    cin >> text;

    for(int nPos = 0; nPos < MAX_TEXT_LENGTH; nPos++)
    {
        if(text[nPos] == '\0') //end of the text
        {
            break;
        }
    }

    //votre code ...
    // ...
    int nFreq = 10;
    char mot[] = "ABA";

    cout << nFreq << "\r\n";
    cout << mot << "\r\n";
    return 0;
}
```