

Programmation Impérative - TP 1

Tableaux et Chaînes de caractères

Rohan Fossé - Léo Mendiboure - Guillaume Mercier
{rohan.fosse,leo.mendiboure}@labri.fr, mercier@enseirb-matmeca.fr

2019-2020

1 Échauffement

Écrire un programme qui calcule le nombre de caractères de chaque type présent dans une chaîne (de caractères) passée au programme en argument. Les majuscules et les minuscules ne seront pas distinguées. On utilisera un *unique tableau* pour stocker cette information.

Par exemple si la chaîne `toTotAta` est passée au programme, celui-ci affichera :

```
a: 2
o: 2
t: 4
```

2 Chiffrement

Vous allez écrire un programme avec deux modes de fonctionnement :

- Mode chiffrement. Le programme transforme la chaîne de caractère (lue sur l'entrée standard) en fonction de la clé de chiffrement (donnée en argument) de la manière suivante :
 - les lettres 's' seront remplacées par 3 points
 - les lettres 'o' seront remplacées par 3 tirets
 - toutes les autres lettres (majuscules et minuscules) seront chiffrées par décalage (code de César). Le décalage étant donné par la clé de chiffrement.
 - les autres caractères resteront inchangésLe résultat est affiché sur la sortie standard.
- Mode déchiffrement. Le programme fait la transformation inverse de celle décrite dans le mode chiffrement.

Le mode de fonctionnement est sélectionné par un argument spécial passé en ligne de commande : `-c` pour le mode chiffrement et `-d` pour le mode déchiffrement. Le mode est le *premier* des arguments et la clé de chiffrement est le *second*. La chaîne de caractères à traiter est récupérée sur l'entrée standard par la redirection de l'entrée du programme sur un fichier.

Fonctionnement attendu :

```
$cat fichierAcrypter
>Le marchand de légumes est arrivé

$./prog -c 2 < fichierAcrypter
>Ng octejcpf fg néiwog... g...v cttkxé

$./prog -c
```

```
>2 arguments au minimum : mode clé_de_chiffrement
```

```
$/prog tata titi
```

```
> Veuillez préciser un mode.
```

3 Aide aux mots fléchés

Dans cette partie vous allez implémenter un programme d'aide aux mots fléchés. Le programme a pour but d'afficher les mots possédant une lettre donnée à une position donnée et ayant une certaine taille.

Un fichier servant de dictionnaire sera redirigé vers l'entrée standard du programme lors de l'appel de ce dernier. Il est à récupérer ici : `/net/ens/mercier/PG109/TP1/dictionary.txt`. De plus,

- le premier argument sera la position de la lettre dans le mot à trouver
- le deuxième argument sera la lettre à trouver dans le mot
- le troisième argument sera la taille du mot à trouver

Dans notre cas l'appel du programme sera par exemple :

```
./aideFleches 2 b 5 < dictionary.txt
```

Exercice :

Écrivez une fonction qui lit sur l'entrée standard le contenu du dictionnaire (vous utiliserez la fonction `fgets`, cf le manuel) à l'aide d'une boucle. On arrête de lire quand la fin de fichier est atteinte (cf manuel de `fgets`). Vous analyserez le mot courant *à la volée* et ne l'afficherez que s'il correspond aux critères de recherche.