

Écriture de fonctions

**En langage C, le passage des paramètres à une fonction se fait toujours par valeur !**

**Reprenez le programme écrit au TP 9 : « Chaîne de caractères et fonction "fgets" ».**

**1. Fonction qui reçoit des paramètres et ne renvoie aucune valeur**

- Ajoutez une fonction
  - PRE : `t` : tableau de chaînes de caractères de taille `sz` ne contenant pas de chaînes nulles.
  - POST : Les chaînes de caractères de `t` sont affichées sur `stdout` dans l'ordre des indices à raison d'une chaîne par ligne.
  - `void impTable (char ** t, int sz)`
- Modifiez le programme de telle manière qu'il fonctionne comme suit :
  - il récupère les mots sur la ligne de commande,
  - les met en table,
  - affiche la table des chaînes de caractères grâce à la fonction `impTable`
  - tant que la fin de fichier n'est pas rencontrée
    - lit un mot d'au plus 256 caractères
    - regarde si le mot est contenu dans la table
    - affiche le mot suivi de « présent » ou « absent » suivant le cas
  - affiche le nombre de fois qu'un mot lu n'a pas été trouvé dans la table.

**2. Fonction qui reçoit des paramètres et renvoie une valeur**

Ecrivez une fonction

- PRE : `inv` : chaîne de caractères  
`s` : tableau de char de taille `sz`
- POST : La fonction a affiché le message d'invitation `inv` sur `stdout`, ensuite il a lu sur `stdin` une chaîne de caractère d'au plus `sz - 1` caractères, a vérifié qu'elle n'est ni vide, ni trop longue et répété l'opération tant qu'une chaîne valide n'est pas introduite, puis il a remplacé le `'\n'` par `'\0'`. La chaîne valide a été placée dans `s`.
- RES : renvoie le nombre de caractères de la dernière chaîne lue ; -1 si fin de fichier [Ctrl-D] atteinte
- `int litEtValideChaine (char* inv, char* s, int sz)`

Modifiez le programme afin que chaque lecture soit faite grâce à la fonction `litEtValideChaine`.

### 3. *Code refactoring*

Cet exercice est relatif à l'exercice « Puissance NxM » de la séance 6. Vous pouvez soit repartir de votre solution, soit repartir de la solution que nous avons mise à votre disposition.

Vous avez sans doute remarqué que le code source de la fonction *main* est assez long, voire très long. Nous vous demandons de réorganiser votre code de manière à le rendre plus lisible. Une manière d'y parvenir est d'introduire des fonctions faisant une partie du travail.

Concrètement, nous vous demandons de « factoriser » votre code en y introduisant au moins les fonctions suivantes : *afficherPlateau*, *lireColonne*, *placerPion*, *ajouterColonne*, *supprimerColonne*.

### 4. *Spécifications*

Ecrivez les specs des fonctions que vous avez conçues au point 3.