INF203 - Travaux pratiques, séance 6

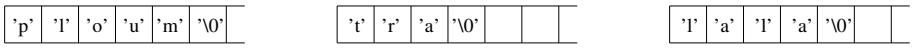
C : Chaines de caractères, structures

Pour s'échauffer : quelques gammes avec des chaînes de caractères

`[TP6]` • Dans le fichier `chaines.c`, complétez la fonction `mon_strlen` pour calculer la longueur de la chaîne passée en paramètre.

• Dans le fichier `chaines.c`, ajoutez une fonction de profil
`void mon_strcpy(char *destination, char *source)`
qui copie la chaîne `source` dans la chaîne `destination`.

[a]



Dessinez comme ci-dessus la chaîne obtenue en concaténant les 3 chaînes `"ploum"`, `"tra"`, et `"lala"`. ■

• Dans le fichier `chaines.c`, ajoutez une fonction de profil
`void mon_strcat(char *destination, char *source)`
qui concatène la chaîne `source` à la fin de la chaîne `destination`. On suppose que la mémoire allouée à la chaîne `destination` est suffisante.

• Dans le fichier `chaines.c`, ajoutez une fonction de profil
`int mon_strcmp(char *chaine1, char *chaine2)`
qui compare les chaînes `chaine1` et `chaine2` et renvoie :
— `0` si les chaînes sont identiques,
— `1` si les chaînes diffèrent.

[b] Quels tests avez vous effectués pour vérifier que vos fonctions sont au point ? Joignez le listing de `chaines.c` à votre compte-rendu. ■

Exercice complémentaire :

> Modifiez votre fonction `mon_strcmp` pour qu'elle ait un comportement similaire à celui de `strcmp` de la bibliothèque standard, c'est-à-dire pour qu'elle renvoie :
> — `0` si les chaînes sont identiques ;
> — un entier négatif si `chaine1` précède `chaine2` dans l'ordre lexicographique (l'ordre du dictionnaire) ;
> — un entier positif si `chaine1` suit `chaine2` dans l'ordre lexicographique.

Lorsque ces 4 fonctions sont au point, vous gagnez le droit d'utiliser les fonctions de la bibliothèque standard `strlen`, `strcpy`, `strcat` et `strcmp`.
Si nécessaire, pensez à ajouter la ligne `#include <string.h>` au début de vos programmes pour y avoir accès.

Fabriquer des phrases

Le principe :

Dupliquez votre programme `chaines.c` en `phrases.c`, puis supprimez le contenu de la fonction `main`. Ajoutez (dans la fonction `main`) les déclarations suivantes (ou toute autre variante) :

```
char Sujet[50]="la petite souris" ;
char Verbe[50]="mange" ;
char Compl[50]="le gros chat" ;
char Phrase[150] ;
```

Puis à l'aide des fonctions de copie et de concaténation de chaînes, écrivez les instructions permettant de fabriquer la phrase complète dans la chaîne `Phrase`, puis de l'afficher.
[c] Avez-vous pensé à ajouter des espaces pour séparer les mots ? Si non, faites-le ! ■

Cadavre exquis

Nouvelle modification du `main`. En utilisant la fonction `generer_entier` comme lors du TP5, générez des phrases aléatoires en concaténant les chaînes des tableaux définis dans le fichier `bouts_de_phrases.c`, en prenant garde de bien séparer les mots par des espaces. Votre programme générera une dizaine de phrases et les affichera.

• *Récréation :* Modifiez les contenus des tableaux à votre convenance.

• Écrivez une fonction `nb_mots` ayant en paramètre une chaîne de caractères représentant une phrase comme celles que vous avez généré, et qui renvoie le nombre de mots de cette phrase.
[d] Quel caractère comptez-vous pour compter les mots ? ■

Utilisez cette fonction pour afficher le nombre de mots de chaque phrase générée.

Sacs de billes

Prenez connaissance du fichier `billes.c` (et exécutez le programme).
Ajoutez une fonction `void afficher_joueur_V2(joueur* pj)` qui affiche les informations concernant le joueur `*pj`. Vous pouvez utiliser les notations `pj->pseudo` et `pj->nb_billes` à la place de `(*pj).pseudo` et `(*pj).nb_billes`.

Ajoutez une fonction `lire_joueur(joueur* pj)` permettant de lire depuis le clavier et de mémoriser les informations concernant le joueur `*pj`.
[e] Pourquoi le profil de cette fonction ne pouvait-il pas être `lire_joueur(joueur j)` ? ■

Essayez quand-même : écrivez une fonction `transferer_V1(joueur j1, joueur j2, int nb)` dans laquelle `nb` billes sont transférées de `j1` à `j2`. Vérifiez que ça ne marche pas !
Écrivez une nouvelle version `transferer_V2` avec un autre passage de paramètre, qui elle, fonctionne ...
[f] Joignez `billes.c` à votre compte-rendu. ■

La commande de la semaine : `tr`

Exécutez la commande `tr aeiou vwxyz` et tapez quelques lignes de texte. Observez les réponses affichées par la commande.

Terminez « proprement »les entrées de cette commande à l'aide des touches `Ctrl`+`D`.

On rappelle également que `tr -s abc` remplace chaque suite de `a`, de `b` ou de `c` par une unique occurrence de ce caractère.

Expérimentez cette commande sur le fichier `exemples_tr` qui vous est fourni.

[g] Quelle commande tapez-vous pour créer une copie de `fich1` nommée `fich2`, dans laquelle toutes les lignes vides ont été supprimées ? ■