

Programmation C

TP 8 - Les strings

I - Objectifs

Nous allons créer un **petit jeu de type pendu**. L'objectif est de **deviner un mot caché** (ex: -----). Pour cela le joueur peut proposer des lettres. Si une lettre est bien présente dans le mot alors elle est placée sur le mot caché (ex: --LL-). Si le joueur propose une lettre qui ne se trouve pas dans le mot alors il perd une vie.

Nous allons simplifier un peu les règles pour ce TP.

- Le mot a deviné sera écrit en dure dans le code (ex: LOL).
- Pas de mot composé (pas d'espaces).
- Le joueur ne peut que proposer des lettres une par une (pas de proposition du mot entier).
- Le joueur commence avec 10 points de vie.

...

Proposer une lettre > P

Non la lettre 'P' n'est pas présente dans le mot "---", il vous reste 9 vies

Proposer une lettre > L

Oui la lettre 'L' est bien présente dans le mot "L-L"

Proposer une lettre > O

Bravos, vous avez trouve le mot "LOL" et il vous reste 9 vies

...

II - Préparatifs

- 1- Créer un nouveau projet C du nom de **"tp8_string"**.
- 2- Remplacer le texte "hello world" par "TP8: Les strings".

III - Instructions

1- Créer les variables et constantes nécessaires au programme (mot_secret, saisie_utilisateur, ...).

2- Créer la boucle de jeu

2.1- Demander au joueur de faire une proposition de lettre (attention, utiliser la commande "fflush(stdin);" avant le scanf pour ne pas risquer de lire des restes entré au clavier).

2.2- Si la proposition est bonne remplacer les '-' correspondants par la lettre. Sinon faire perdre une vie au joueur.

3- Gérer la fin de partie. Victoire: Cas où le joueur a trouvé toutes les lettres ou Défaite: cas où le joueur n'a plus de vies.

IV - Pour aller plus loin

A- Offrir la possibilité, au joueur, de soit donner une lettre, soit proposer directement le mot entier si il pense avoir deviné le bon mot.

V - SPOIL! (aide)

1- Il faut créer les variables suivantes

- `int nb_vie=10`, `char saisie_utilisateur`
- le tableau qui contient la chaîne secrète: `char mot_secret[] = "PROGRAMMATION";`
- la taille de la chaîne secrète: `int taille_mot = strlen(mot_secret);`
- le tableau des char trouvés: `char mot_trouve[taille_mot+1];` (Attention au + 1 dans la taille du tableau pour avoir une case pour le `'\0'`)
- Enfin il faut initialiser ce tableau avec des `'-'` et ne pas oublier le `'\0'` dans la dernière case

2.1- Utiliser `fflush(stdin)` puis un simple `scanf` car ici, il faut lire un simple char (attention au `&` devant votre variable char dans le `scanf`)

2.2- Il faut parcourir le tableau contenant le mot secret et vérifier pour chaque case si la saisie utilisateur est identique. Si le char est présent dans le mot secret, on l'ajoute au tableau `"mot_trouve"` sinon on fait perdre une vie.

3- Il faut faire une boucle `do while` et vérifier que le nombre de points de vie est supérieur à 0. Pour la victoire, il faut comparer la chaîne trouvée avec la chaîne du mot secret via `strcmp`. si elles sont identique c'est gagné ! il ne faut pas oublier de forcer la sortie de la boucle de jeu via un `break`;

A- Il faut remplacer la saisie utilisateur par un tableau de char et utiliser la fonction `fgets` au lieu du `scanf`. ainsi il est possible, pour l'utilisateur, d'écrire soit un char soit une string.

Il faut séparer le cas où l'utilisateur rentre une lettre et le cas où il en rentre plusieurs (proposition d'un mot)