

Projet C : Motus

Avant de commencer : la qualité des commentaires, avec notamment la présence des antécédents, des conséquents, des invariants de boucle, les rôles de chacune des fonctions, ainsi que les noms donnés aux variables, l'emploi à bon escient des majuscules et la bonne indentation rentreront pour une part importante dans l'appréciation du travail. Ce projet doit permettre de montrer votre autonomie et votre compréhension tant dans la conception du programme que dans sa réalisation. Enfin, si les codes de plusieurs projets se trouvent être identiques, ou être copiés depuis le web, tous les projets concernés seront immédiatement sanctionnés par un zéro.

Objectif : faire une application qui permet à un utilisateur de jouer à **Motus**, célèbre jeu de la télévision française. Voir [fr.wikipedia.org/wiki/Motus_\(jeu_t%C3%A9l%C3%A9vis%C3%A9\)](https://fr.wikipedia.org/wiki/Motus_(jeu_t%C3%A9l%C3%A9vis%C3%A9)).

1 Le jeu

Le jeu consiste à trouver des mots de 7 à 10 lettres à partir d'une lettre initiale fixée, et cela en un nombre de tentatives limité à 6. Les mots sont pris dans un dictionnaire de mots communs. À chaque tentative, l'utilisateur propose un mot du dictionnaire de même longueur que le mot recherché (sinon la proposition est nulle) et en retour on lui indique, pour chaque lettre, si elle est à la bonne place, ou à une autre place, ou n'est pas présente dans le mot.

2 Première version : interface textuelle

Dans un premier temps, votre application *motus*, interférera avec le joueur *via* l'entrée et la sortie standard.

L'application choisit au hasard le mot à rechercher dans un dictionnaire de mots de 7 à 10 lettres. Vous utiliserez obligatoirement le dictionnaire disponible le site Moodle/LMS (voir plus loin).

Le joueur propose des mots, au maximum 6. Pour chacune des lettres d'un mot proposé, l'application indique :

- 'x' : si la lettre est à la bonne place;
- 'o' : si la lettre est présente dans le mot mais pas à sa place;
- '.' : si la lettre n'est pas présente dans le mot.

Si le mot proposé n'est pas dans le dictionnaire ou bien n'a pas le nombre de lettres attendus, la précédente proposition est conservée.

Ci-dessous un exemple d'exécution de l'application.

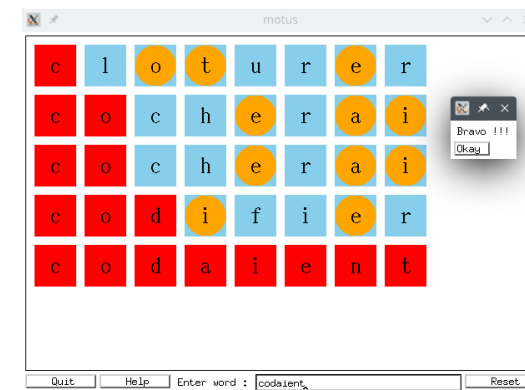
```
c-----
--> cloturer
cloturer
x-oo--o-
--> cocherai
cocherai
xx--o-oo
--> conclurez
```

```
cocherai
--> codifier
codifier
xxxo--o-
--> codaient
codaient
xxxxxxx
Bravo, vous avez gagné
```

3 Seconde implémentation : Interface graphique

On veut maintenant visualiser graphiquement le jeu. Pour cela, vous utiliserez la bibliothèque graphique *libsx*. L'utilisateur saisira ses propositions de mots par l'intermédiaire d'une `StringEntry`, et les réponses seront visualisées dans une `DrawArea` selon les codes couleurs du jeu télévisé : rouge pour les lettres bien placées, orange pour les lettres mal placées et bleu pour les lettres qui ne font pas partie du mot.

Si on reprend l'exemple précédent, où le mot recherché était *codaient*, votre interface graphique pourrait ressembler à :



4 Remise du projet

Votre projet est à faire en binôme (voir liste des binômes) et vous devrez le rendre au plus tard :

le dimanche 9 juin 2024, 23h59 – aucun délai ne sera accordé –

sous forme d'une archive `motus.tgz` que vous déposerez sur le site de dépôt **Moodle/LMS UCA 23 : EIEL622 - ECUE Langage C-2**. **Ne déposer qu'un seul projet par binôme.**

Attention : ce site gère la date limite de rendu. Après ce sera trop tard. N'attendez pas la dernière minute, vous pouvez déposer des versions intermédiaires de votre projet.

Cette archive devra contenir :

- les fichiers `.h` et `.c`, correctement commentés (chaque fonction/procédure doit avoir un commentaire, les invariants de boucle doivent être marqués), indentés, et codés (les noms de variables explicites, évitez les trop longues fonctions/procédures).
- le fichier `Makefile` ;
- un fichier `Documentation`, **exclusivement au format pdf**, décrivant le fonctionnement général du programme et les originalités du code ;
- la compilation avec les options `-Wall` ne doit pas donner de *warning*.

Attention de bien respecter toutes les directives indiquées dans ce projet. Si tel n'est pas le cas, la note s'en ressentira. Commencez à travailler ce projet dès à présent !

Have fun.