

Projet C : Pendu (le jeu)

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 (pas de chatgpt), tous les projets concernés seront immédiatement sanctionnés par un zéro.

Objectif : faire une application munie d’une interface graphique qui permet à un utilisateur de jouer au *Pendu*, célèbre jeu d’écoliers.

Vous trouverez sur wikipédia à l’adresse [https://fr.wikipedia.org/wiki/Pendu_\(jeu\)](https://fr.wikipedia.org/wiki/Pendu_(jeu)) une description de ce jeu avec ses règles.

Ci-dessous, les contraintes de programmation que **vous devez absolument respecter** :

1. utilisation exclusive du langage C ;
2. l’interface graphique doit être programmée exclusivement avec la bibliothèque graphique *libsx* ;
3. respect des consignes de structuration d’une application munie d’une interface graphique comme vues en cours (*i.e.* indépendance entre le moteur de l’application et l’interface graphique) ;
4. votre interface graphique doit aussi posséder les boutons :
 - « Rejouer » pour lancer une nouvelle partie ;
 - « Aide » pour fournir une aide en ligne aux utilisateurs.
5. vous devrez structurer votre programme à l’aide de routines réparties de façon cohérente dans plusieurs modules (fichiers *.c* avec leurs *.h*) ;
6. la compilation est faite à l’aide d’un Makefile ;
7. la longueur des mots à chercher est comprise entre 6 et 8 caractères ;
8. le pendu est formé de 6 parties, donc le joueur a, au maximum, 6 tentatives pour trouver le mot. Toutefois, ce nombre doit pour être modifié ;
9. utilisation exclusive des fichiers de mots fournis :
 - *dico_fr* (français) ;
 - *dico_uk* (anglais).
10. votre application doit permettre le changement de dictionnaire par l’intermédiaire d’un menu ;
11. vous mémoriserez tous les mots de 6 à 8 caractères pris dans le fichier de mots dans une liste structure simplement chaînée dynamique que vous avez programmée en TD. Le type des éléments de la liste doit être déclaré **#define void* T** ;
12. le mot à chercher est tiré au hasard dans la structure dynamique ;
13. à l’exécution aucune entrée/sortie doit se faire sur E/S standard ;
14. le projet est à faire en binôme (voir liste des binômes).

1 Remise du projet

Votre projet est à rendre au plus tard :

le jeudi 19 juin 2025, 23h59 – **aucun délai ne sera accordé** –

sous forme d’une archive `pendu-nom1-nom2.tgz` que vous déposerez sur le site de dépôt **Moodle/LMS UCA 24 : EIESE643 - ECUE Langage C avancé**. **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 la structure et fonctionnement général du programme et les originalités du code ;
- la compilation avec les options `-Wall` ne doit pas donner de *warning*.

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

Have fun.