

Premier exercice : - Le jeu de pendu

Pour découvrir le jeu: Jeu de pendu : [https://fr.wikipedia.org/wiki/Pendu_\(jeu\)](https://fr.wikipedia.org/wiki/Pendu_(jeu))

Description :

Dans le cadre de ce premier exercice, nous utiliserons un dictionnaire de mots sous la forme d'un fichier texte utilisant uniquement des caractères de la table d'encodage ASCII (pas de caractères accentués par exemple).

Ce fichier dictionnaire contient un mot par ligne, une ligne commençant par le caractère '#' désigne un commentaire et doit être ignorée lors du chargement des mots.

Les mots ne doivent contenir que (au minimum 1, au maximum 26) des caractères alphabétiques, en majuscule ou minuscule, pour être plus pratique les majuscules seront converties en minuscules lors du chargement dans le dictionnaire.

Le joueur doit deviner un mot en proposant un caractère, le jeu lui indique ou est ce caractère (il peut être présent plusieurs fois) dans le mot à deviner. Le joueur dispose d'un nombre de tentatives limité.

Travail attendu :

0- prendre en main le code fourni (fichier .h et de test)

1- vous devez **définir, et justifier la structure de données** qui permet de stocker toutes les lettres déjà devinées par le joueur :

- nature de la structure de données,
- portée et durée de vie de cette variable.

2- Vous devez coder les fonctions suivantes (voir le détail dans le fichier fonction.h) :

int estMotValide(char mot[]);	Difficulté algorithmique : ★☆☆☆	Difficulté technique : ★☆☆☆
int chargerMots(char chemin[]);	Difficulté algorithmique : ★☆☆☆	Difficulté technique : ★★☆☆
int lireCaractere();	Difficulté algorithmique : ☆☆☆☆	Difficulté technique : ★★☆☆
int estTrouve(char mot[MAXLEN]);	Difficulté algorithmique : ☆☆☆☆	Difficulté technique : ★★☆☆
int main(int argc, char *argv[]);	Difficulté algorithmique : ★☆☆☆	Difficulté technique : ★☆☆☆

3- Passer les **tests** fournis et vérifier les résultats. Éventuellement enrichir les **jeux de tests**.

4- Générer à partir des **commentaires du code**, la documentation automatique via le wizard **doxygen**, pour cela vous devrez étudier les différentes options de configuration.

5- Être capable de réaliser une **démonstration de votre programme et une explication** de votre code.