**Projet 1: Jeu du pendu**

**Objectif**

Réalisation du jeu du pendu.

**Présentation du jeu**

Il se joue à deux. L'un pense à un mot, par exemple :

|  |
| --- |
| CAHIER |

Et écrit une suite de tirets à la place des lettres :

|  |
| --- |
| ------ |

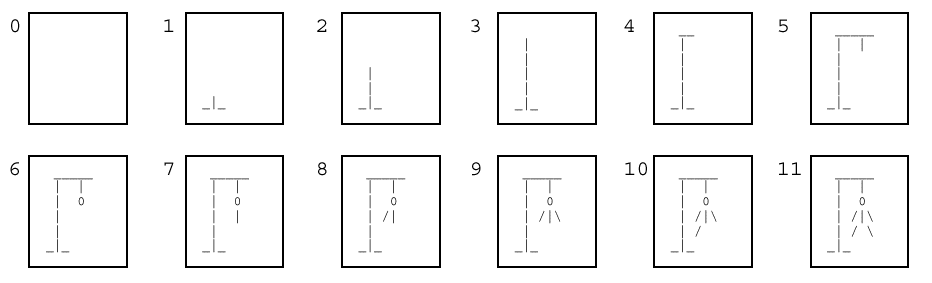
L'autre doit deviner ce mot en ne connaissant au départ que sa longueur. Pour s'aider, il peut demander l'existence d'une lettre. A chaque lettre proposée, si celle-ci figure dans le mot, les occurrences de la lettre sont écrites à leur place. Par exemple, pour E :

|  |
| --- |
| ----E- |

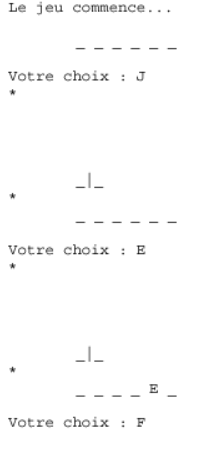
Si la lettre ne figure pas dans le mot, le joueur est pénalisé d'un point. Au bout de sept pénalisations, le joueur est < pendu > a moins qu'il ne trouve le mot avant.

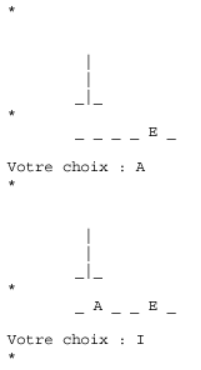
En général le jeu est agrémenté d'une sinistre potence dont le malheureux pendu se dessine progressivement à chaque mauvaise proposition.

Voici l'avancement du dessin du pendu en fonction du nombre d'erreurs du joueur :

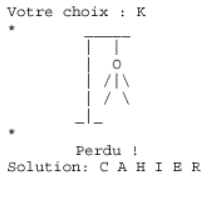
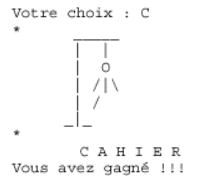


Voici l'affiche à fournir à l'utilisateur :



[ ….]



|  |  |
| --- | --- |
| **en cas de défaite** | **en cas de victoire** |

**Déroulement du programme :**

La machine organise le jeu.

1. Le premier joueur entre un mot de 3 à 10 lettres pendant que l'autre joueur ne regarde pas.
2. Le mot s'affiche avec des blancs soulignés a la place des lettres.
3. Le programme demande la saisie d'une lettre.
4. Le second joueur tape une lettre.
   1. En cas de succès, le mot s'affiche (avec les lettres trouvées).
   2. En cas d'échec, le programme indique combien de propositions il reste.
5. Le mot s'affiche avec des lettres et des soulignes selon qu'elles aient été découvertes ou non.
6. Le programme reprend au point 3 sauf si le joueur a épuisé tous ses coups ou trouve le mot. A la fin de la partie, le programme affiche si le joueur est pendu ou non.

**Travail demandé :**

Réaliser le Jeu du pendu en utilisant la langage Python.