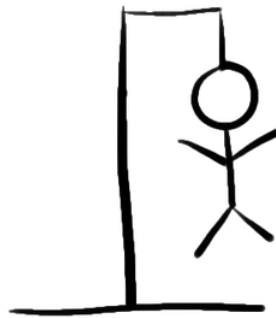


INFO-F101 – Programmation

Projet 2

Le pendu

Année académique 2017–2018



Le pendu est un jeu consistant à trouver un mot en devinant quelles sont les lettres qui le composent. Le jeu se joue traditionnellement à deux, avec une feuille et un crayon, selon un déroulement bien particulier. Quand le dessin est terminé on voit un bonhomme allumette qui se pend¹.

Règles

Chaque partie commence par la sélection d'un mot à deviner. Une fois qu'un mot est choisi, le nombre de lettres à deviner est indiqué par un tiret bas (_) pour chaque lettre. Le joueur peut alors proposer une lettre. Si la lettre se trouve dans le mot à deviner, toutes les occurrences de la lettre sont remplies à leurs positions respectives (qui étaient données par les tirets bas). Si la lettre choisie ne se trouve pas dans le mot à deviner, le dessin du pendu commence par l'apparition de la potence. La prochaine lettre choisie qui ne se trouve pas dans le mot donne lieu à l'apparition de la tête, etc. Le jeu est perdu quand le dessin est complet et le jeu est gagné quand toutes les lettres ont été données.

Modalités

Nous vous demandons d'implémenter ce jeu en python3 et de permettre à un joueur de deviner un mot. Un fichier devra contenir les mots à deviner, dont un est choisi au hasard au début de chaque jeu. Afin de dessiner le pendu et de demander au joueur une lettre, vous ferez usage de la librairie `turtle` (la documentation est disponible à <https://docs.python.org/3.6/library/turtle.html>), dont voici un exemple d'utilisation :

1. [https://fr.wikipedia.org/wiki/Le_pendu_\(jeu\)](https://fr.wikipedia.org/wiki/Le_pendu_(jeu))

```
>>> import turtle
>>> turtle.goto(300,30)
>>> turtle.circle(15)
```

Votre programme commencera par lire le fichier qui contient les mots à deviner (un mot par ligne) et en choisira un au hasard. Le fichier s'appellera *mots.txt*. Votre programme fera une liste de tous les mots et en sélectionnera un en utilisant la fonction `randint` de la librairie `random`. Initialement un tiret bas est dessiné pour chaque lettre composant le mot à deviner dans la fenêtre graphique (`turtle`). Hors les tirets bas, la fenêtre est vide au début du jeu. Ensuite, le joueur pourra entrer une lettre grâce à une fenêtre `turtle` qui permet de faire des input. Si un joueur introduit plus qu'une lettre, le programme doit redemander un input valable. Si la lettre donnée se trouve dans le mot à deviner, toutes les occurrences de cette lettre deviennent visibles dans la fenêtre graphique à leur position respective. Si la lettre donnée ne se trouve pas dans le mot à deviner, cette lettre apparaît dans la fenêtre graphique de façon distinguable des lettres composant le mot et le dessin du pendu progresse par l'inclusion de la potence. Pour chaque lettre donnée qui ne se trouve pas dans le mot, le dessin continue à se compléter (la séquence est donnée dans figure 4). Le jeu est perdu quand le dessin est complet et le jeu est gagné quand toutes les lettres ont été données.

Le programme doit être exécutable via la commande suivante aux salles machines du bâtiment NO :

```
~>python3 projet2.py
```

Voici un exemple d'exécution du programme attendu :

Après sélection du mot à deviner de façon aléatoire la fenêtre graphique indique le nombre de lettres comprenant le mot et une fenêtre `turtle` permet d'introduire une lettre (figure 1). Dans cet exemple le mot est *programmation*.

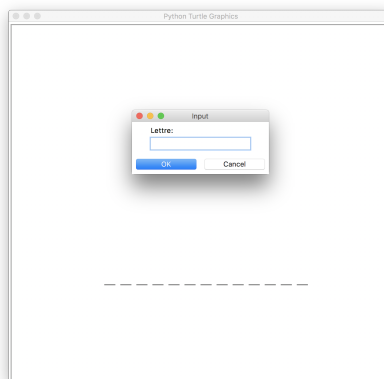


FIGURE 1 – Fenêtre graphique initiale.

Le joueur peut alors introduire une lettre. Si la lettre se trouve dans le mot, toutes les occurrences de la lettre sont remplies. Dans l'exemple la lettre *o* est introduite (figure 2).

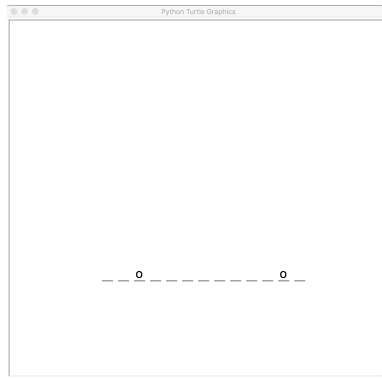


FIGURE 2 – Fenêtre graphique initiale.

Quand toutes les lettres sont remplies, la partie est gagnée (figure 3).

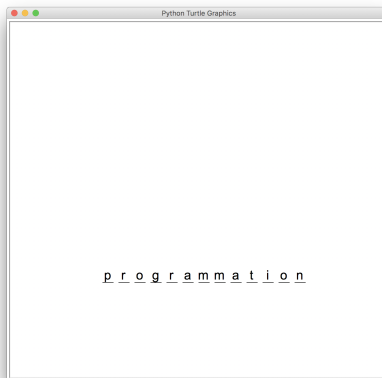
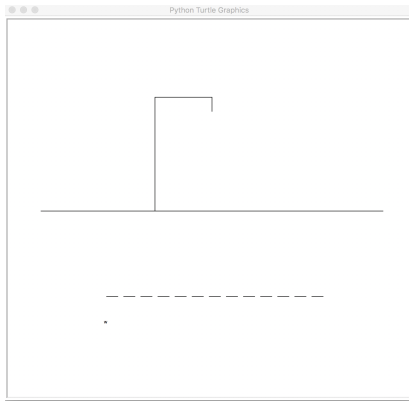
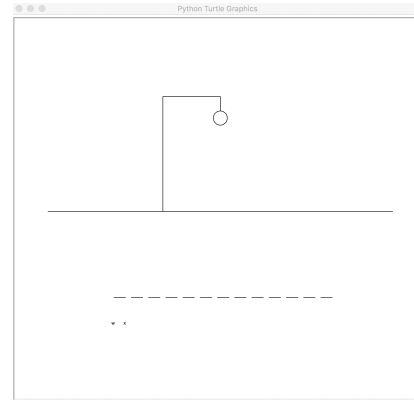


FIGURE 3 – Une partie gagnée.

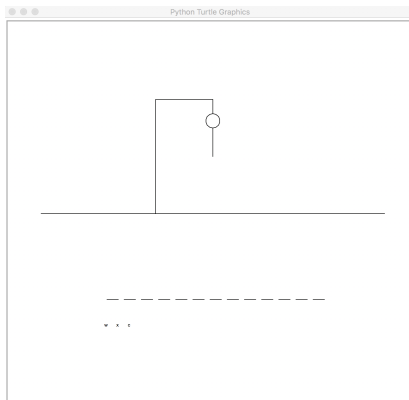
Dans le cas que la lettre ne se trouve pas dans le mot, le dessin du pendu progresse comme illustré dans les figures 4 -.



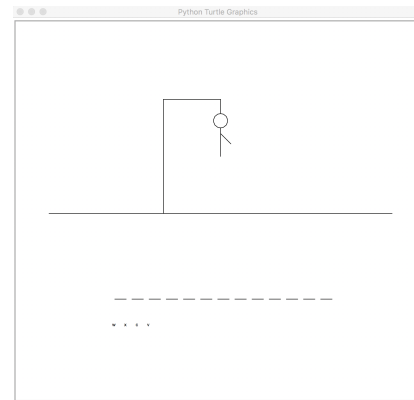
(a) Première erreur.



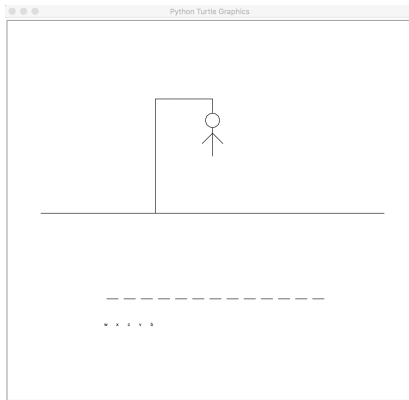
(b) Deuxième erreur.



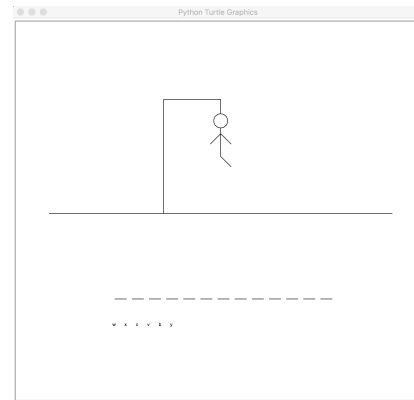
(c) Troisième erreur.



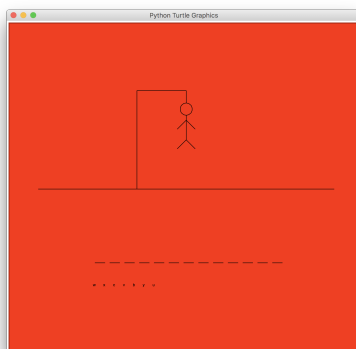
(d) Quatrième erreur.



(e) Cinquième erreur.



(f) Sixième erreur.



(g) Septième erreur.

FIGURE 4 – Progression du pendu.

Veillez à commenter votre programme avec pertinence.

Conseil

Ce projet contient deux aspects ; le dessin du pendu et la fonctionnalité qui permet de deviner le mot.

Consignes pour la remise du projet

Le projet devra être remis via deux canaux : une version imprimée au secrétariat étudiant (Maryka Peetroons - P.2N8.104) et une version électronique sur l'Université virtuelle. Pour le document imprimé, n'utilisez pas de farde et/ou chemise en plastique. S'il comporte plusieurs pages, agrafez-les. Veillez à soigner la présentation de ce document. Même imprimé en noir et blanc, utilisez la coloration syntaxique afin d'en améliorer la lisibilité et évitez les retours à la ligne non indentés pour vos commentaires. Les consignes pour la remise du projet sont disponibles en ligne sur la page du cours sur l'Université Virtuelle. Les consignes sont à respecter *scrupuleusement* ; relisez-les attentivement avant la remise !

Votre projet sera testé à l'aide de la commande suivante : `python3 projet2.py`.

Pour toute question concernant l'énoncé, nous vous invitons à vous adresser à Nassim Versbraegen (nversbra@vub.ac.be).

Date limite de remise. Le lundi 30 octobre 2017 à 13h.