

Mini-Projet: Jeu de Nîmes

I) Introduction

a) But du jeu

Notre programme est le jeu de Nîmes.

Un certain nombre de bâtons sont disposés sur le plateau de jeu. Chacun son tour, chaque joueur pourra prendre jusqu'à 3 bâtons d'un coup. Le perdant est celui qui prend le dernier bâton.

b) Pourquoi ce jeu nous intéresse-t-il ?

C'est un jeu facile à comprendre et facile à développer.

II) Cahier des charges

Paramétrage

Le programme sera personnalisable et demandera :

1. Le nom du J1 et du J2. Cependant, leurs noms seront stockés dans une liste. L'id du J1 sera donc 0, car il correspond au premier élément de la liste.
2. Le nombre de bâtons voulu ($2 < x < 21$ Par défaut : 10)

Jeu

Le joueur qui commence sera choisi au hasard.

Le jeu se déroulera dans cet ordre :

1. Afficher les bâtons
 2. Demander à J1/J2 de tirer un 1 à 3 bâtons d'un coup
 3. Enlever X bâtons ($0 < X < 4$)
 4. Vérifier s'il reste un bâton
- Si oui : Fin du jeu (cf. plus bas)
 - Si non : Le jeu continue --> Recommencer pour le joueur suivant

Fin du jeu

Il est affiché :

- Bravo J1/J2
- Nombre de tours
- Le temps qu'ils ont pris

Spécificités techniques

Variables utilisées

- Nom du J1 : `nom_J1`
- Nom du J2 : `nom_J2`
- Liste des noms (afin: `noms`)
- Nombre de bâtons : `nbr_batons`
- Nombre de tours : `nbr_tours`
- Nombre de bâtons à enlever : `input_joueur`
- Heure de départ/heure de fin : `heure_depart` / `heure_fin`

Représentation du plateau de jeu

Le nombre de bâtons sera stocké dans une variable nommée `nbr_batons`.

Initialisation

Le nombre de bâtons sera stocké avant le jeu.

Les paramètres seront choisis, le jeu sera initialisé puis une boucle `while` sera initialisée, elle se terminera à la fin du jeu (voir Réalisation>Difficultés rencontrées).

Affichage des bâtons

Nous afficherons les bâtons restants sur une même ligne grâce au caractère ASCII 124 (Barre verticale) : |

Enlever des bâtons

On soustrait le nombre de bâtons enlevés au nombre de bâtons présents: `nbr_batons = input_joueur - nbr_batons`

Il faudra faire attention à ne pas enlever plus de bâtons qu'il en reste.

Il faudra vérifier si le Joueur n'enlève pas plus de bâtons qu'il n'en reste.

Techniques choisies

Le programme sera écrit en Python (UTF-8).

Le fichier sera stocké sur GitHub afin d'assurer un partage de fichier sans problème.

Un terminal sous Bash permettra l'exécution du programme.

Choix de l'éditeur de texte Visual Studio Code pour écrire le code source en Python.

Nous utiliserons les modules suivants :

- `choice` du module `random` (Pour choisir le joueur qui commence)
- `datetime` (Pour le temps pris pour jouer) et `timedelta` (Calculer la différence entre l'heure de début et de fin) du module `datetime`

Style

Nous utiliserons le Snake Case (style conventionnel de Python) pour les variables.

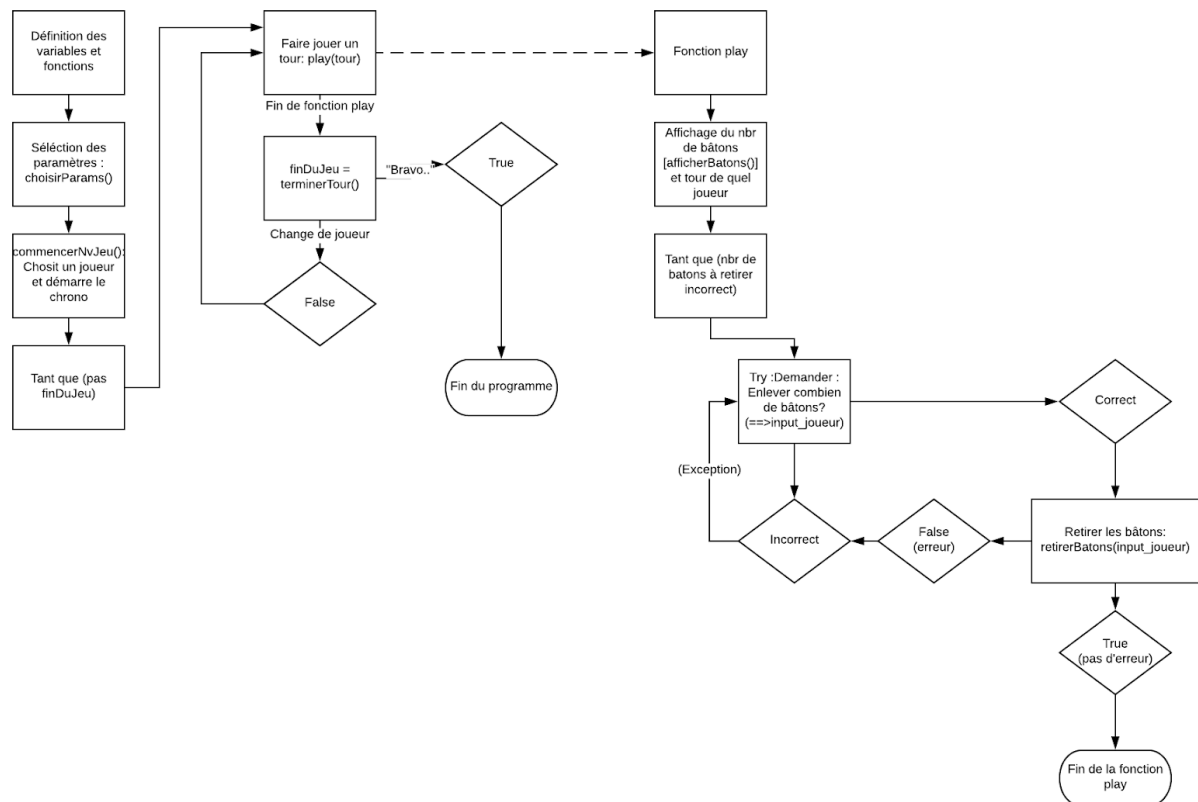
Cependant, nous utiliserons le Camel Case pour le nom des fonctions.

Les fonctions sont documentées selon le style Docstring de Google.

III) Réalisation

Le programme est séparé dans de nombreuses fonctions : il n'y a que 3 grandes instructions dans le code qui appellent toutes les autres fonctions. Il est vrai que déclarer autant de fonctions ne semble pas nécessaires. Cependant, cela est volontaire car nous pensons que cela permet au code d'être plus modulable si nous voulons implémenter de nouvelles fonctionnalités.

Voici le diagramme fonctionnel de notre programme :



Plus de précisions sont à trouver dans le code source.

Nous avons rencontré certaines difficultés :

Calculer le temps de jeu:

Nous avons eu besoin d'Internet pour savoir comment utiliser le module `datetime`.

Nous stockons l'heure de départ lorsque le jeu commence (à la fin de l'appel de la fonction `commencerNvJeu`, après avoir choisi les paramètres). A la fin du jeu, nous stockons une nouvelle fois l'heure et on fait la différence. L'heure s'affiche donc de la manière suivante : `{minutes} mins` et `{secondes} secondes`

Récurtivité

Au départ, la fonction `play` été appelée après `terminerTour`. Le problème est que, par conséquent, la fonction `play` s'appelait elle-même, ce qui engendre une récursivité non voulue qui peut s'avérer nocive si nous voulons améliorer notre programme plus tard. Comme c'est une boucle infini, il est très difficile de déboguer.

IV) Planification:

La répartition du travail était assez inégal, on a tous les 2 fait le cahier des charges mais le code est en grande partie fait par Killian. Nous avons rencontrés quelques soucis pour se voir car les événements actuels nous ont pas aidés.

V) Réalisation:

Le cahier des charges a été réalisé en 2 heures, le code a pris 4 heures à être réalisé. Le jeu a été codé de manière correcte et marche comme nous l'avons voulu. Nous aurions aimé demander aux joueurs s'ils voulaient faire une nouvelle partie.

Nous avons été fiers de concevoir ce code qui marche.

VI) Fonctionnement

1er cas:

Il reste 3 bâtons et je souhaite en retirer 3 :

Tour n° : 8

C'est au tour de J1

3 bâtons restants

|||

Combien de bâtons voulez-vous retirer? (1 à 3) : 3

Vous ne pouvez pas enlever tous les bâtons! Vous pouvez enlever 1 à 2 bâton

PlusDeBaton

Vous avez retiré trop de bâtons

2eme cas:

Je ne peux retirer qu'un seul bâton:

Tour n° : 9

C'est au tour de J2

2 bâtons restants

||

Combien de bâtons voulez-vous retirer? (1 à 3) : 3

Vous ne pouvez pas enlever tous les bâtons! Vous pouvez enlever 1 bâton uni

PlusDeBaton

Vous avez retiré trop de bâtons

3eme cas:

Je souhaite mettre plus ou moins de bâton que ce qui est possible:

Entrez le nombre de bâtons ($2 < x < 21$; défaut: 10): 21

Nombre de bâtons non valide

Nombre de bâtons : 10

Nous aurions aimé redemander le nombre de bâtons à enlever au lieu de forcer le démarrage de la partie.

4eme cas:

Je souhaite retirer plus ou moins de bâton que ce qui est possible:

Tour n° : 2

C'est au tour de J1

18 bâtons restants

|||||||

Combien de bâtons voulez-vous retirer? (1 à 3) : 9

Interval

Chiffre non compris entre 1 et 3!

Combien de bâtons voulez-vous retirer? (1 à 3) : 3

5eme cas:

j'entre une chaîne de caractère à la place de chiffre:

Tour n° : 1

C'est au tour de J1

10 bâtons restants

|||||||

Combien de bâtons voulez-vous retirer? (1 à 3) : toto

invalid literal for int() with base 10: 'toto'

Ce n'est pas un chiffre!

Combien de bâtons voulez-vous retirer? (1 à 3) :

6eme cas:

Je n'entre rien lorsque que l'on me demande le nom des joueurs (==> Nom par défaut) et aucun chiffre lorsque l'on me demande le nombre de bâtons:

```
Entrez le nom du J1 :  
Entrez le nom du J2 :  
Entrez le nombre de bâtons ( 2<x<21 ; défaut: 10):  
Nombre de bâtons non valide  
Nombre de bâtons : 10
```

```
Tour n° : 1  
C'est au tour de J1
```

```
10 bâtons restants  
|||||||
```

VII) Amélioration

- Afficher des couleurs sur le terminale (cela demande plus de temps à implémenter sur Windows).
- Dans le cas où le nombre de bâtons n'est pas valide, ne pas forcer le démarrage de la partie.
- Meilleure gestion de la portée des variables

VIII) Conclusion

Avec ce mini-projet, nous avons appris tout d'abord à travailler à 2, à faire face aux difficultés ensemble et les corriger en proposant des solutions. Le résultat final est

satisfaisant bien que nous aurions aimés y apporter quelques améliorations.

Bilan personnel

Nous avons eu quelques soucis pour réaliser le projet car j'étais absent la 2ème semaine, et Nils la 3ème puis ensuite nous n'avons pas eu cours, j'ai donc dû commencer à coder seul cette 3ème semaine, et cela nous a laissé peu de temps pour répartir les tâches. Pour le prochain projet, nous prendrons plus de temps à élaborer le Cahier des Charges et ainsi répartir le travail. Cependant, je pense que nous avons tous les deux autant travaillé car Nils a été très présent, même lors de la programmation car il m'aidait à déboguer et faisait attention au contenu du code autant que moi. Je suis content de travailler avec Nils car il est très investi et nous sommes tous les deux très motivés.