

# TD - PROBLEMES COMPLEXES

Le projet a été hébergé sur un github, voici l'adresse :

<https://github.com/julien-leroy/Michel-complex-genetique>

Tester le programme afin de constater la qualité du programme :

Lancez le programme python en effectuant la commande : "python main.py".

Il vous sera demandé le nombre de genes , d'individus par générations, de générations ainsi que la methode de sélection utilisé (RWS ou Tournament).

Vous pourrez à ce moment constater que le programme est parfaitement bien codé et constater la présence de la moyenne, l'écart type et de la variance.

Choix faits dans la construction du code :

Nous avons créer des classes afin de gérer en objet chaque parties intégrantes :

- l'essai (classe monde)
- la génération (classe generation)
- l'individu (classe individu)

Chacune de ces classes contient les attributs et le fonctions qui sont propres a leur comportements (exemple : c'est le role de la génération de créer sa population en créant des enfants de la population précédante)

Voici les différents blocks et leur objectif dans leur milieu naturel :

- L'inistialisation : C'est tout simplement la création du premier groupe d'individus, la génération premiere. c'est un groupe d'individus composés de genes (des nombres aléatoires de 0 à 100)
- le calcul des scores : La fonction de fitness permet de calculer le score d'un individu ce qui permettra de la comparé sa capacité a survivre face aux autres individus de sa génération.
- la sélection de deux parents : C'est une sélection aléatoire qui tand vers des individus ayant une bonne chance de survie (score plutot élevé) afin de sélectionner des parents qui produirons une progéniture de qualité. nous avons implémenté 2 méthodes de sélection differentes :
  - Tournament : La technique du tournament sélectionne 2 individus pour confronter leurs score et choisir le pere. La meme sélection est faite pour la mere.

+++ : Nous avons rajouter dans cette fonction la sélection RWS lors de la confrontation des deux individus sélectionné aléatoirement, ce qui rajoute encore de l'aléatoire dans la sélection.

- RWS : La technique RWS consiste à choisir aléatoirement dans une liste d'individus dont le pourcentage de chance d'être choisi dépend de leur fitness. C'est la technique de la roue.
- la copulation : Je pense que le nom est clair... c'est la fonction qui va créer 2 enfant avec les 2 parents choisis grace a la méthode de sélection
- la mutation : C'est une petite chance (10%) qu'un gene mute pour un individu. le gene est aléatoirement choisi et il est juste recréé avec un nombre aléatoire

La variance décrite dans notre programme sert à constater que le coté aléatoire à tendance à ce disciper au fur et a mesure car les scores de nos générations se rapprochent et ont tendance à s'élever vers le haut.

Nous vous avons mis la vitesse d'exécution dans le programme, c'est grace a cela que nous avons constaté une exécution plus rapide avec la sélection tournament plutot que RWS. Par contre, le calcul de la variance ralentis énormément l'exécution du programme

En espérant que vous allez apprecier notre projet, vous avons pris beaucoup de plaisir a le réaliser. Bon courage pour les corrections.

```
groupe = "Walrand Degeorges" + " " + "Julien Leroy")
```

```
print(groupe)
```