
	Structure de données abstraites	Durée : 2H
	TD : Combat de Pokémons	
T NSI		
Centre d'intérêt :		
<b>But de la séquence</b> : Programmer un petit jeu avec 2 dictionnaires.		
<b>Objectif de formation</b> :		
Pré requis : Algorithmique, langage python		
<b>Matériel nécessaire</b> : PC personnel avec l'environnement Anaconda installé.		
Critères d'évaluation :		

## 1. Contexte du jeu

Il s'agit de réaliser un combat de Pokémons et de déterminer lequel en sortira vainqueur.

Chaque Pokémon est défini par 6 caractéristiques dont 4 essentielles pour le combat.

Les caractéristiques :

- Nom : Chaîne de caractères
- Points de vie (HP): Nombre entier naturel
- Puissance d'attaque : Nombre entier naturel
- Vitesse de combat : Nombre entier naturel
- Type : Liste de chaîne de caractères
- Efficacité de défense : Nombre entier naturel

Chaque Pokémon est constitué de caractéristiques différentes mais dont la somme (des nombres) est 200 au maximum. Point à vérifier avant chaque combat.

La méthode "random" sera utilisée pour définir les dégâts à chaque tour :

- attaque = Puissance d'attaque \* random.
- défense = Efficacité de défense \* random.

Une fonction calculDegats() retourne le nombre de points de vie (HP). Le 1<sup>er</sup> paramètre passé est la valeur d'attaque de l'attaquant, le 2<sup>ème</sup> paramètre la valeur de défense du défenseur.

Le nombre HP est > 0 si attaque > défense, 0 dans le cas contraire.

A chaque duel dans un combat, le Pokémon qui attaque est celui qui défendait précédemment.

Déroulement d'un combat : Pokémon1 contre Pokémon2

- Le Pokémon attaquant en premier est celui qui est le plus rapide, sauf dans 20% des cas où c'est le contraire.
- Tant que les points de vie de chaque Pokémon sont > 0
  - ⇒ Si Pokémon2 attaquant : Calcul des HP restant du défenseur (Pokémon1) suivant les dégâts causé par Pokémon2.
  - Sinon : Calcul des HP restant du Pokémon2 suivant les dégâts causé par l'attaquant Pokémon1.
  - ⇒ Changement d'attaquant
  - ⇒ Affichage des points de vie restant de chaque Pokémon.
  - ⇒ Attente de l'appui sur la barre d'espace du clavier pour poursuivre les duels.
- Déclaration du vainqueur

## 2. Réalisation du jeu



Activité 1 : Importer la méthode random de la bibliothèque random, et définir les 2 Pokémon Carapuce et Evoli en créant 2 dictionnaires. Carapuce est de type "Eau", Evoli de type "Normal".

Activité 2 : Réaliser la fonction Calculdegats(Poke1, Poke2) qui retourne le nombre de HP infligés.

Activité 3 : Réaliser la fonction Combat(Poke1, Poke2) qui gère le combat et déclare le vainqueur

Activité 4 : Reporter les résultats de chaque Duel et le résultat final.