**Rapport de TP IA74**

Travail de **TCHAPPI OSEE BRAYAN**

1. L’algorithme produit de solution avec d’énormes fitness mais des solutions pas acceptables.
2. Modifications de l’algorithme
   1. Les modifications sont visibles dans le fichier ***TD1-Question 2-a***. La modification apportée est sur la fonction fitness. Ainsi à la place de mettre un fitness nul pour des solutions non acceptables, nous mettons un fitness négatif et proportionnel au dépassement de la capacité du sac. Cette approche permet donc d’affaiblir les solutions non acceptables.
   2. Les modifications sont visibles dans le fichier **Question2.b**. La modification apportée consiste à « corriger » chaque nouvelle solution de sorte qu’elle acceptable. L’algorithme de cette solution est :

*Corriger\_individu*

*Entrées : individu, capacite\_max, poids*

*Début :*

*Si somme des poids des objets pris <= capacite\_max*

*Retourner individu*

*Sinon*

*Taille = individu.longueur*

*Ind[taille]=[0..0]*

*Indices = tableau contenant des valeurs de 0 à taille-1 ordonnées de manière aléatoire \\ exemple si taille 5 🡪 [3,0,1,2,4]*

*Indice,poids\_total = 0,0*

*Tant que (éléments non parcourus dans indices et poids de l’élément + poids\_total <= capacite)*

*Ind[element] = individu [element]*

*Poids\_total = poids\_total + poids de l’élément*

*Indice ++*

*Fin tant que*

*Individu = ind*

*Fin si*

*Retourner Individu*

*Fin*

Pour ce qui est des comparaisons entre les deux options, les résultats des performances se trouvent dans le fichier : **Comparaison. En guise de conclusion l’algorithme 2.b produit de bien meilleur solution mais prend plus de temps que l’algorithme 2.a.**

1. Adaptation au problème de portefeuille

Les modifications se trouvent dans le fichier question 3. L’adaptation du code a été pensée comme suit :

* Les poids sont assimilés aux prix d’achat des actions
* Les objets deviennent des actions
* La capacité du sac est le budget
* Les valeurs des objets sont les valeurs des titres

Les algorithmes ajoutés sont :

***Indice\_multiple\_action****: renvoie l’id du titre dont plusieurs actions ont été prise et en cas de deux titres achetés en plusieurs actions (possible lors du croisement) corriger en réduisant les actions du titre rapportant le moins*

*Entrées : individu*

*Début*

*Indices = indice des valeurs > 1 d’individu*

*Si indices.longueur > 1 alors*

*Individu [indice du titre rapportant le moins] = random (0,1)*

*Retourner indice du titre qui rapporte le plus*

*Sinon*

*Si indice.longueur = 1 alors retourner indice*

*Retourner -1*

*Fin*

***Vérifier\_risque :*** *permet de vérifier si le titre ayant plusieurs actions respecte la contrainte de ne pas dépasser les 20% du budget.*

*Entrées : prix, individu, budget*

*Début*

*I =* ***Indice\_multiple\_action (individu)***

*Si individu[i]> 0.2\*budget alors retourner Faux*

*Retourner Vrai*

*Fin*

La correction des individus est faite suivant l’algorithme suivant :

***Correction\_individu***

*Entrées : individu, budget, prix\_d’achat*

*Début :*

*Si budget non dépassé et* ***Vérifier\_risque*** *= vrai*

*Retourner individu*

*Taille = individu.longueur*

*Ind[taille]=[0..0]*

*Indices = tableau contenant des valeurs de 0 à taille-1 ordonnées de manière aléatoire \\ exemple si taille 5 🡪 [3,0,1,2,4]*

*Indice,poids\_total = 0,0*

*Si* ***Indice\_multiple\_action (individu) = -1)*** *// pas de titre ayant plusieurs actions achetées*

*Taux\_multi\_action = 0.75 // à faire varier au bon vouloir*

*Tirer une valeur aléatoire*

*Si (valeur tirée <= taux) alors*

*Sélectionner aléatoirement un titre*

*Affecter une valeur d’actions entre 1 et nombre max d’action tel qu’on ne dépasse pas les 20% du budget*

*Fin si*

*Tant que (éléments non parcourus dans indices et poids de l’élément + poids\_total <= capacite)*

*Si element >1*

*Tant que element\*prix du titre > 0.2\*budget faire*

*Element –*

*Fin si*

*Ind[element] = individu [element]*

*Poids\_total = poids\_total + poids de l’élément*

*Indice ++*

*Fin tant que*

*Retourner Ind*

*Fin*

1. Les résultats se trouvent dans le fichier question4