

COMPTE-RENDU TP4 : **Algorithmes Génétiques**

Algorithmes des différentes fonctions :

lecture_gene : Entrée : Serpent | Sortie : Tableau de valeurs

Principe : Lis le génome du serpent passé en entrée dans un tableau de valeurs où l'expression y est écrite (en prenant en compte les gènes d'opérateurs)

- Initialisation de *gene_tab* en tant que tableau de « nombre de gène - 1 » valeurs
- Pour *index* allant de 0 à (*nombre de gène* / 2)
 - 4 premiers bits de la valeur positionnée à la position *index*, convertit en entier et placé dans *gene_tab* à la position $2 * index$
 - identification des deux derniers bits : Si 00, alors '+' | Si 01, alors '-' | Si 10, alors '*' | Si 11, alors '/' => ajout du résultat dans *gene_tab* à la position $2 * index + 1$
- Retourne *gene_tab*

calcul : Entrée : Serpent | Sortie : 0 si Maléfique trouvé, sinon 1

Idée : La difficulté dans cet algorithme est de gérer les priorités dans l'expression du score du serpent. « *lecture_gene* » lit et traduit l'expression dans un tableau de valeurs. Dans ce tableau de valeurs, les calculs prioritaires y seront effectués. On parcourt les opérateurs dans le tableau de valeurs et dès qu'on tombe sur un '*' ou un '/', l'opération effectuée entre l'opérande précédent l'opérateur et l'opérande suivant l'opérateur. Puis on remplace l'opérande précédent par le résultat et on décale toutes les valeurs suivant le 2^{ème} opérande de 2 cases « vers la gauche » (écrasant l'opérateur et le 2^{ème} opérande). Une fois toutes les multiplications et divisions effectuées, on effectue à la suite les différentes additions et soustractions, suivant le tableau (sans se préoccuper du décalage) pour pouvoir avoir à la fin le résultat de l'expression. Le score du serpent sera $|Résultat - 666|$.

- Initialisation :
 - *tab_gene_lu* = tableau de valeurs (initialisé avec *lecture_gene()* du serpent)
 - *Résultat*
 - *Save*
 - *verif_erreur* = 0 (registre de vérification en cas d'erreur = division par 0)
 - *index_VIDE* = position dernière valeur = nombre de gène - 2
 - *gene_index* = 1 (position du 1^{er} opérateur dans *tab_gene_lu*)
- Tant que *gene_index* < *index_VIDE* & *verif_erreur* != 1
 - Si l'opérateur lu est '*' ou '/'
 - Si '/'
 - Si l'opérande suivant vaut 0, alors *verif_erreur* = 1
 - Sinon, *save* = opérande précédent / opérande suivant
 - Sinon *save* = opérande précédent * opérande suivant
 - affectation : opérande précédent = *save*

- Pour *index_tab* allant de *gene_index* + 2 à *index_VIDE*, valeur à la position *index_tab* est décalée de 2 cases vers la gauche
- Valeur *VIDE* affectée aux cases d'index *index_VIDE* et *index_VIDE* - 1
- *index_VIDE* se décrémente de 2.
- Si l'opérateur lu en *gene_index* (après décalage) n'est ni '*', ni '/', alors *gene_index* s'incrémente de 2
- *gene_index* réaffecté à 1
- Affectation : *Résultat* = 1ère valeur du tableau
- Tant que *gene_index* < *index_VIDE* & *verif_erreur* != 1
 - Si l'opérateur lu en *gene_index* est '+', *Résultat* = *Résultat* + *opérande suivant*
 - Si c'est un '-', *Résultat* = *Résultat* - *opérande suivant*
 - Incrémentation de *gene_index* de 2
- Si *verif_erreur* = 1, alors *Score du Serpent* = *MAX*
- Sinon, *Score du Serpent* = |*Résultat* - *OBJECTIF*|

évaluation : Entrée : Population | Sortie = Registre Maléfique

Principe : Il s'agit de calculer le score de chaque serpent de la population et ainsi renvoyer le registre qui confirme ou infirme la présence ou non d'un serpent maléfique. La fonction triera aussi les serpents en fonction de leur score. De plus, la moyenne et l'écart-type des scores des serpents de la population seront calculés.

- Initialisation :
 - *verif_maléfique* = 1
 - *calcul_moyenne* = 0
 - *calcul_variance* = 0
 - Moyenne, Écart-type
- Pour *membre_index* allant de 0 à *population_nombre*
 - calcul du score du serpent membre d'index *membre_index*
 - Si le score est de 0, *verif_maléfique* = 0 ;
 - Ajout du score à *calcul_moyenne* (addition)
- Tri *BubbleSort* en fonction du score des serpents
- *Moyenne* = *calcul_moyenne* / *population_nombre*
- Pour *membre_index* allant de 0 à *population_nombre*,
 - $\text{calcul_variance} = \text{calcul_variance} + (|\text{Score du Serpent} - \text{moyenne}|)^2$
- $\text{écart-type} = (\text{calcul_variance} / \text{population_nombre})^{1/2}$
- Afficher *Moyenne* et *Écart-type*
- Retourner *verif_maléfique*

Des fonctions ont été ajoutées pour pouvoir simplifier le code total.
Vous les retrouvez, ainsi que les fonctions demandées, dans le fichier *fonctions.c*

Merci pour votre lecture