

Corrigé type

Questions de connaissance.

Q1. voir le cours **1pt**

Q2. voir le cours **1pt**

Q3. Citer les principes de base pour une recherche heuristique ?

- Intensification (ou exploitation) : permet d'examiner en profondeur une zone particulière de l'espace de recherche **0.5pt**
- Diversification (ou exploration) : permet d'orienter la recherche vers de nouvelles zones (prometteuses) dans l'espace de recherche **0.5pt**

Q4.

• Méthode exacte = fournit une solution exacte en explorant systématiquement l'espace de recherche. **0.5pt**

Méthode approchée = fournit une solution approchée en explorant partiellement l'espace de recherche. **0.5pt**

• Méthode déterministe = fournit le même résultat pour le même input (y a pas d'actions probabilistes). **0.5pt**

Une méthode indéterministe = fournit différents résultats pour le même input (y a des actions probabilistes). **0.5pt**

• Heuristique = méthode approchée (ou non) spécifique à un problème déterminé. **0.5pt**

Métaheuristique = méthode générique applicable à tout problème. **0.5 pt**

Q5. Instances de problèmes NP-difficiles dont la taille de l'input est large. **0.5pt**

Q6. Modélisation dans le type des genes. (PG val des genes = des actions, AG val des genes = valeur binaire, decimales) **0.5 pt**

Q7. - liste tabou trop large \Rightarrow plus d'efficacité \checkmark et plus de temps CPU \times . **0.5 pt**

- liste tabou trop restreinte \Rightarrow moins d'efficacité \times et moins de temps CPU \checkmark . **0.5pt**

Correction1.

1. Le problème consiste à choisir une classe pour chaque cours.

- Variables : C_1, \dots, C_k (les différentes classes pour chacun des cours) **0.5pt**
- Domaine : $\{1, \dots, c\}$ **0.5pt**
- Contraintes : Pour tout les i, j , si $(E_i \geq E_j > S_i \text{ OU } S_i \leq S_j < E_i)$ alors $(C_i \neq C_j)$ **1pt**
 - $S_i < E_i, S_i > 1$ (pour 1 :00), $E_i < 5$ **1pt**

2. Les difficultés d'un PO

- Taille de l'espace de recherche **0.5pt**
- Paysage de la fonction de coût **0.5pt**

Correction 2.

Population : 0.5pt

De taille n individus

Initialisation de l'algorithme génétique : 0.5pt

Initialement, les n individus (chromosomes) sont générés de manière aléatoire

Chromosome (individu) :

Codage (de 1 à 9)

Le chromosome est une matrice de dimension 9×9 ou un vecteur de longueur 9×9

Fonction objectif à minimiser (Fitness function) : 1pt

Il s'agit de compter le nombre de chiffres qui ne sont pas correctement placés.

Mutation : **0.5pt**

Choisir avec une probabilité p_m une case de manière aléatoire et inverser sa valeur

Croisement : **0.5pt**

Croisement à 1-point avec une probabilité p_c

Choisir un point de croisement et échanger les parties des deux individus

Sélection : **0.5pt**

Elitiste ou la roue de la fortune (Roulette Wheel)

Critère d'arrêt : **0.5pt**

S'arrêter lorsque les 9 chiffres sont bien placés.

Ou bien on atteint un nombre maximum MaxGen d'itérations

Correction 3.

1. **2pts**

Fonction RechercheTabou ()

$A^* \leftarrow$ Génère solution initiale ()

ListeTabou $\leftarrow \emptyset$

Pour iter = 1 à Max Mouvements Faire

 Sélection du meilleur dans voisinage(A) excepté voisins dans ListeTabou

 MiseAJour(ListeTabou)

 Si $f(A) < f(A^*)$ Alors $A^* \leftarrow A$

retourner $A^*, f(A^*)$

/* Creation et initialisation de la liste tabou */ **0.5pt**

/* procedure de création de l'ensemble de voisinage avec evaluation de fonction d'adaptation */ **0.5pt**

Toutes les reponses justes sont acceptés

2.

Recherche Tabou : améliorations

• Liste tabou – Réglage dynamique de la longueur de la liste tabou **0.5pt**

- à augmenter si solutions fréquemment répétées (diversification)

- à diminuer en cas d'absence d'améliorations (intensification)

- réglage aléatoire périodique parmi [Lmin, Lmax]

• Critère d'aspiration – Autoriser des mouvements tabou qui mènent à une meilleure solution **0.5pt**