Examen (Session principale)

Filière : GLSI
Niveau : Deuxième année
Matière : Fondements de l'intelligence artificielle
Enseignants: R. GUESMI, M. SAKKARI

A.U : 2021-2022
Date : 00-00-2022
Durée : 00H
Documents : non autorisés

Exercice 1 : 8 puzzle problème utilisant Hill-climbing (6 pts)

- Soit un plateau 3×3 avec 8 tuiles (chaque tuile a un numéro de 1 à 8) et un espace vide. L'objectif est de placer les numéros sur des tuiles pour correspondre à la configuration finale en utilisant l'espace vide. Nous pouvons glisser un tuile à gauche, à droite, au-dessus et en dessous dans l'espace vide.

:

2	8	3
1	6	4
7		5

Configuration initiale

1	2	3
8		4
7	6	5

Configuration finale

- Soit la fonction objectif f(c)=nombre de tuiles bien placés + n. Le niveau de configuration n est incrémenté de 1 à chaque passage.

Question 1 (5 points):

Donner la suite des configurations selon la méthode hill-climbing de **c** parcourues.

★ n=1 :

	actuelle, n=1	possibles (c'est-à-dire niveau suivant),n=2		
	c=1	c=2 **?**	c=3 **?**	c=4 **?**
Configuration	2 8 3 1 6 4 7 5	2 8 3 1 4 7 **6** 5	2 8 3 1 6 4 **7** 5	2 8 3 1 6 4 7 **5**
f(c)	4+1=5	**5+2=7** ☑	3+2=5	3+2=5

➔ suite des configurations de **c** parcourues : 1

★ n=2 :

	actuelle, n=2	possibles (c'est-à-dire niveau suivant),n=3
	c=2	à déterminer
Configuration	2 8 3 1 4 7 6 5	à déterminer
f(c)	7	à déterminer

➔ suite des configurations de **c** parcourues : 1→2

★ n=3 …….

Question 2 (1 point): Quelle configuration **c** trouverait la méthode hill-climbing?

Exercice 2 : MiniMax (3 pts)

Question 1 (1 point): Compléter les phrases suivantes à l'aide des mots choisis dans la liste ci-dessous.

- la racine, terminal, profondeur d'abord, le plus court chemin, le coup parfait,largeur d'abord, déterministe, admissible, discret

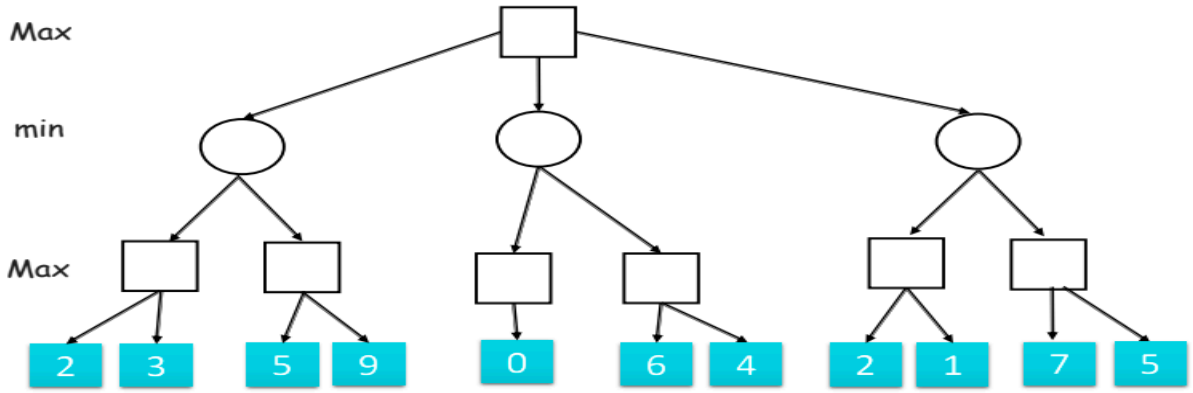
Un jeu à deux joueurs est défini classiquement comme un arbre qui a comme noeuds des positions. Chaque noeud est un noeud «joueur» ou un noeud «opposant».

➢ Si un noeud n'a pas de fils, c'est un noeud ……

➢ l'objectif de la méthode MiniMax est de trouver …… pour un jeu …… à information parfaite.

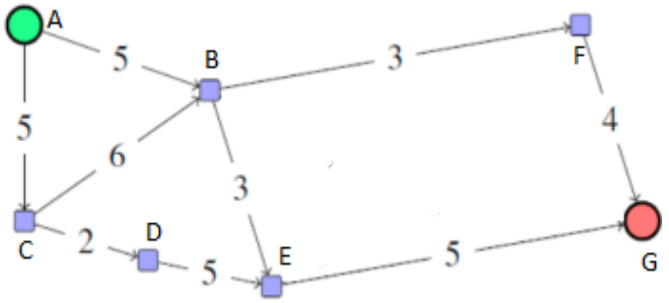
➢ l'algorithme MiniMax utilise la recherche en ……

Question 2 (2 points): donner la trace d'exécution (état final seulement) de l'agorithme MiniMax pour l'exemple suivant.Quelle résultat peut-on déduire ?



Exercice 3 : A* pour la recherche dans un graphe (11 points)

Considérez la carte suivante. L'objectif est de trouver le chemin le plus court de A vers G. On donne également trois heuristiques, h1, h2 et h3.



Noeud	A	B	C	D	E	F	G
h1	10	5	10	10	5	3	0
h2	10	8	11	6	2	5	0
h3	10	6	11	9	2	4	0

Soit la classe Graph suivante :

```
# 1-Définir notre graph
class Graph:
    # Initialiser la classe
    def __init__(self, graph_dict=None, directed=True):
        self.graph_dict = graph_dict or {}
        self.directed = directed
    ...
    def connect(self, A, B, distance=1):
        self.graph_dict.setdefault(A, {})[B] = distance
        if not self.directed:
            self.graph_dict.setdefault(B, {})[A] = distance
    ...
```

<u>Questions :</u>

1. Donner les instructions qui permettent :
   - de créer les listes open et closed.(0.5 point)
   - de créer le nœud de départ et le nœud objectif, sachant que le constructeur **`def __init__(self, name:str, parent:str)`**, est utilisé pour créer un objet Node. (1 point)
   - d'jouter le nœud de départ à la liste open.(0.5 point)
   - de créer un objet Graph. (0.5 point)
   - de créer les connexions de ce graphe (1.5 point)
   - de créer les heuristiques pour chaque node. (1 point)
2. Est-ce que h1, h2 et h3 sont admissibles ? Justifier. (2 points)
3. Quelles relations de dominance existent entre ces trois heuristiques? (1 point)
4. Appliquer la recherche A* en utilisant h1. Donner la suite des noeuds développés. (3 points)

Bon travail