

Série 3

Exercice 1:

1. • DFS:Ordre: $S \rightarrow A \rightarrow C \rightarrow D \rightarrow E \rightarrow G$ Chemin final: $S \rightarrow A \rightarrow C \rightarrow D \rightarrow E \rightarrow G$ Coût total: $4 + 1 + 5 + 4 + 3 = 17$ • BFS:Ordre: $S \rightarrow A \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow G$ Chemin final: $S \rightarrow C \rightarrow E \rightarrow G$ Coût total: $2 + 5 + 3 = 10$ • UCS:Ordre: $S \rightarrow C \rightarrow B \rightarrow D \rightarrow E \rightarrow G$ Chemin final: $S \rightarrow C \rightarrow E \rightarrow G$

Coût total: 10

• A*:

Ordre:

On explore S et on prend le chemin $S \rightarrow C$ On explore C et on prend le chemin $S \rightarrow C \rightarrow E$ On explore E et on prend le chemin $S \rightarrow C \rightarrow E \rightarrow G$ Chemin final: $S \rightarrow C \rightarrow E \rightarrow G$

Coût total: 10

2. • Admissible: $S \rightarrow G: 2 + 5 + 3 = 10 > 5 = h(S)$ $A \rightarrow G: 1 + 5 + 3 = 9 > 5 = h(A)$ $C \rightarrow G: 5 + 3 = 8 > 4 = h(C)$ $D \rightarrow G: 4 + 3 = 7 > 5 = h(D)$ $E \rightarrow G: 3 > 2 = h(E)$ $G \rightarrow G: 0 = 0 = h(G)$

$B \rightarrow G$: étant donné qu'il n'existe pas de chemin entre B et G, $h(B)=25$ ne dérange pas car cela permet à l'algorithme d'éviter d'aller en B.

Donc c'est admissible.

• Consistent: $S \rightarrow A: h(S)=5 < 4 + h(A)=9$ $S \rightarrow C: h(S)=5 < 2 + h(C)=6$ $S \rightarrow B: h(S)=5 < 8 + h(B)=33$ $A \rightarrow D: h(A)=5 < 3 + h(D)=8$ $A \rightarrow C: h(A)=5 < 1 + h(C)=5$ $C \rightarrow B: h(C)=4 < 2 + h(B)=27$

Donc c'est consistant

 $C \rightarrow D: h(C)=4 < 5 + h(D)=10$ $C \rightarrow E: h(C)=4 < 5 + h(E)=7$ $D \rightarrow E: h(D)=5 < 4 + h(E)=6$ $E \rightarrow G: h(E)=2 < 3 + h(G)=3$ $G \rightarrow B: h(G)=0 < 1 + h(B)=26$ $G \rightarrow C: h(G)=0 < 1 + h(C)=5$

3.

Node	Distance
A	∞
B	∞
C	∞
D	∞
E	3
G	0

On regarde les voisins de G

Node	Distance
A	∞
B	∞
C	8
D	7
E	3
G	0

On regarde les voisins de E

Node	Distance
A	10
B	∞
C	8
D	7
E	3
G	0

On regarde les voisins de D

Node	Distance
A	9
B	∞
C	8
D	7
E	3
G	0

On regarde les voisins de C

Node	Distance
A	9
B	∞
C	8
D	7
E	3
G	0

On regarde les voisins de A

Selon moi, B reste à l'infini car il n'est prédécesseur de personne

4. UCS: il cherche le chemin le moins coûteux entre S et G.
 Dijkstra: il cherche le chemin optimal entre S et tous les noeuds du graphe
 A*: il estime le chemin optimal entre S et G en utilisant le coût du chemin de S à N (qui est un autre noeud) et en ajoutant une heuristique qui estime le coût restant jusqu'à G.
 Si l'heuristique est bonne, A* est plus rapide que Dijkstra