**Fondements de l’intelligence Artificielle**

**EX1 :**

1. \*\*

Etat init : carte non colorer.

But : carte colorer tels que deux pays adjacents non pas le meme couleur.

Fonction successeur : assignez les couleurs à un région ou pays.

Fonction de cout : nombre de région colorer.

1. \*\*

Etat init : le singe est au sol.

But : le singe est au plafond de piéce.

Fonction successeur : le singe déplace vers la caisse suivante.

Fonction de cout : nombre des caisses parcouru.

1. \*\*

Etat init : les récipients sont vides.

But : 1 litre d’eau dans tous les récipients.

Fonction successeur : remplir les récipients ou verser entièrement leur contenu dans un autre récipient ou sur le sol.

Fonction de cout : nombre des foix de déplacement d’eau.

**EX2 :**

1. \*\*
2. \*\*

Non, on n’a pas besoin de savoir la distance.

1. \*\*

BFS

\*\*(a,0,b,0) f{/}

Succ : {(b,0,b,0), (c,0,b,0)}

F : {(b,0,b,0), (c,0,b,0)}

\*\*(b,0,b,0) f{(c,0,b,0)}

Succ : {(c,0,c,0), (a,0,a,0), (b,1,b,0)}

F : {(c,0,b,0), (c,0,c,0), (a,0,a,0), (b,1,b,0)}

\*\*(c,0,b,0)

Succ : {/}

\*\*(c,0,c,0)

Succ : {(b,0,c,0), (a,0,c,0), (c,1,c,0)}

F : {(a,0,a,0), (b,1,b,0), (b,0,c,0), (a,0,c,0), (c,1,c,0)}

\*\*(a,0,a,0)

Succ : {(b,0,a,0), (c,0,a,0), (a,1,a,0)}

F : {(b,1,b,0), (b,0,c,0), (a,0,c,0), (c,1,c,0), (b,0,a,0), (c,0,a,0), (a,1,a,0)}

\*\*(b,1,b,0)

Succ : {/} déjà visité

\*\*(b,0,c,0)

Succ : {/} déjà visité

\*\*(a,0,c,0)

Succ : {/} déjà visité

\*\*(c,1,c ,0)

Succ : {(c,1,c,1)} goal state

F : {(c,1,c,0), (b,0,a,0), (c,0,a,0), (a,1,a,0), (c,1,c,1)}

**EX3:**

1. \*\*
2. \*\*

A/ largeur d’abord:

L’etat initial est 1

R=[1] on arrive en 1

R=[2,3] on arrive on 2

R=[3,4,5] on arrive en 3

R=[4,5,6,7] on arrive en 4

R=[5,6,7,8,9] on arrive en 5

R=[6,7,8,9,10,11] on arrive en 6

R=[7,8,9,10,11,12,13] on arrive en 7

R=[8,9,10,,11,12,13,14,15] on arrive en 8

R=[9,10,11,12,13,14,15] on arrive en 9

R=[10,11,12,13,14,15] on arrive en 10

R=[11,12,13,14,15] on arrive en 11

R=[12,13,14,15]

L’etat 11 est atteint.

L’etat 11 est l’etat finale

B/Profondeur d’abord c:

L’etat initial est 1

R=[1] on visite 1

R=[2,3] on visite 2

R=[4,5,3] on visite 4

R=[8,9,3,5] on visite 8

R=[9,5,3] on visite 9

R=[5,3] on visite 5

R=[10,11,3] on visite 10

R=[11,3] on visite 11

R=[3]

L’etat 11 est atteint

L’etat 11 est l’etat finale

C/Profondeur d’abord limitee a 2 :

L’etat initial est (1,0)

R=[(1,0)] on visite (1,0)

R=[(2,1),(3,1)] on visite (2,1)

R=[(4,2),(5,2),(3,1)] on visite (4,2)

R=[(8,4),(9,4),(5,2),(3,1)] on visite (8,4)

R=[(9,4),(5,2),(3,1)] on visite (9,4)

R=[(5,2),(3,1)] on visite (5,2)

R=[(10,5),(11,5),(3,1)] on visite (10,5)

R=[(11,5),(3,1)] on visite (11,5)

R=[(3,1)]

L’etat 11 est atteint

L’etat 11 est l’etat finale

**EX4:**

L'état initial : #CFLO

L'état final: CFLO#

Séquence 1 : CF#LO

=> Elle correspond à un état valide car le loup ne peut pas manger le chou.

Séquence 2 : CFL#

=> Elle correspond à un état invalide à cause d'un passager (chou O).

Succ(FLO#C) = {LO#CF, O#CFL, L#CFO}