

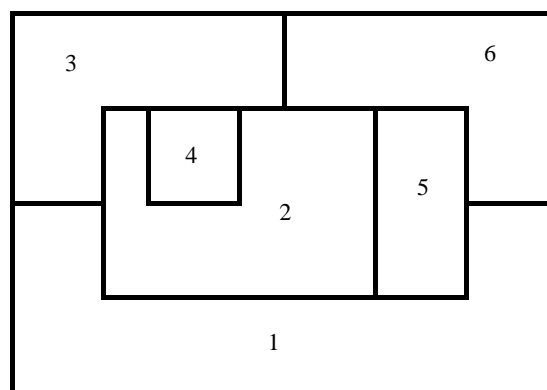
Prolog & IA *Série 4*

Exercice 1

Coloriage d'une carte :

Ecrire une procédure en Prolog qui permet de colorier un graphe non orienté avec une liste de couleurs, de telle sorte que deux sommets adjacents ne soient pas de même couleur.

Appliquer cette procédure pour colorier avec 4 couleurs, la carte suivante :



Donner une autre application de cette procédure pour colorier une carte de pays (ou de régions) de votre choix avec 4 couleurs.


Exercice 2

Le compte est bon :

Partant d'une liste d'entiers et en les recombinaut par les opérations élémentaires (addition, soustraction, multiplication, division exacte), il est possible d'atteindre un nombre donné.

Par exemple :

Avec $L = [25, 7, 4, 2, 3]$ on obtient 347 avec $(25 * (7 * 4) / 2) - 3$.

Avec $L = [25, 5, 2, 3, 4, 10]$ on obtient 1225 avec $25 * ((5 + 2) + (3 * (4 + 10)))$. 

Ecrire une procédure qui donne la liste des opérations nécessaires pour obtenir un entier donné à partir d'une liste de nombres entiers.

Exercice 3

Transvasement :

Soient 3 récipients non gradués de contenances maximales : 3, 5, 8 litres, contenant initialement 0, 0, 8 litres. Comment arriver à la situation 0, 4, 4 litres, sachant qu'on ne peut verser un récipient A dans un récipient B, que jusqu'à ce que A soit vide ou que B soit plein.

Exercice 4

Files de moutons :

Une file de moutons noirs rencontre une file de moutons blancs sur un chemin très étroit, entre les deux files, il y a juste la place nécessaire pour un mouton. Le but est d'arriver à faire se croiser les deux files.

Chaque mouton noir peut avancer d'une case vers la droite, et sauter par dessus le mouton qui est devant lui (une case à droite) si la case où il arrive est libre. Symétriquement, chaque mouton blanc peut avancer d'une case vers la gauche, et sauter par dessus le mouton qui est devant lui (une case à gauche) si la case où il arrive est libre. Les moutons ne peuvent pas reculer.

Exercice 5

Tour de Hanoi :

Comment déplacer une tour formée de N disques de diamètres échelonnés (le plus petit en haut), d'un emplacement dit gauche à un autre dit droite, en se servant d'un troisième (milieu). La règle est de ne jamais placer un disque sur un autre dont le diamètre serait plus petit.

1. Donner les solutions du problème trouvées avec l'algorithme de recherche en profondeur itérée et l'algorithme de recherche en largeur d'abord.
2. Donner une solution avec l'algorithme suivant :

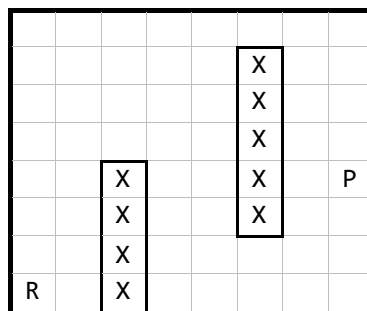
Pour déplacer les N disques de gauche sur droite, il faut déplacer les N-1 premiers sur milieu, puis le dernier (qui est le plus grand) sur droite, et enfin les N-1 qui ont été placés sur milieu, les transporter sur gauche.

Que peut-on conclure ?

Exercice 6

On considère le problème de labyrinthe :

Par exemple, un robot noté R dans une salle de NxM cases où il y a des murs (cases X) et avec une porte noté P :



On se propose de chercher le plus court chemin que peut traverser le robot pour atteindre la porte, sachant que le robot ne peut pas se déplacer en diagonale et qu'il ne peut traverser les murs.

1. Donner une formulation générale du problème.
2. Appliquer l'algorithme A* pour trouver la solution de l'exemple.

Exercice 7

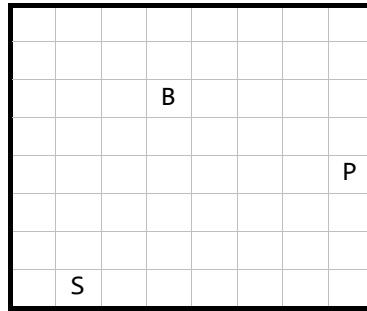
On considère le problème du singe et des bananes :

Dans une salle, il y a un singe, une boîte et un panier de bananes suspendu au plafond.

On se propose d'aider le singe à attraper les bananes, sachant que :

Que, pour attraper les bananes, le singe doit déplacer la boîte au-dessous du panier et monter sur la boîte.

On considère la situation suivante, qui donne un exemple de la dimension de la salle, les positions du singe (S), du panier (P) et de la boîte (B) :



Le singe peut se déplacer vers le sud, l'est, l'ouest ou le nord (il ne peut se déplacer en diagonales), et lorsqu'il est dans une case adjacente à celle de la boîte sur la même ligne (ou sur la même colonne), il peut :

- pousser la boîte en avant ou tirer la boîte en arrière. Les nouvelles positions de la boîte et du singe doivent rester sur la même ligne (ou la même colonne) que celle des anciennes positions (Il ne peut déplacer la boîte en diagonales).
- monter sur la boîte.

En fin, si le singe est sur la boîte, il peut descendre en occupant une case adjacente sur la même ligne (ou la même colonne) que celle de la boîte.

1. Donner une formulation du problème.
2. Ecrire une procédure qui utilise l'algorithme A* et une heuristique (à définir) pour trouver une solution optimale du problème :

Cette procédure doit spécifier les différentes étapes constituant la solution : la suite des actions et les différentes positions du singe et de la boîte.