

**Examen Final**  
Juin 2018

**Coefficient : 40%.**

**Aucun document autorisé.**

**Remarques et conseils :**

- Lisez **attentivement** chaque question avant d'y répondre.
- Indiquez clairement sur votre copie le numéro de l'exercice avant d'y répondre.
- Lorsque vous définissez un prédicat, son profil et sa définition formelle doivent être au moins indiqués. Un jeu d'essais est facultatif.
- Expliquez autant que possible vos choix lors de la définition d'un prédicat ou d'une fonction.
- Le barème défini ci-après est susceptible d'être modifié.

**Exercice 1 : Questions de cours (5 points)**

1. Quelles sont les étapes qui permettent de passer d'une formule quelconque à une formule sous forme standard ?
2. Démontrer à l'aide du principe de résolution la validité ou la contradiction de la formule suivante :  

$$\neg(\exists x \forall y \neg(S(x) \leftrightarrow S(y)) \wedge \forall x S(x))$$
3. Soit le CSP (X,D,C) suivant :
  1.  $X = \{X1; X2; X3; X4\}$  l'ensemble des variables,
  2.  $D = \{D1,D2,D3,D4\}$  leurs domaines de valeurs, avec  $D1=D2=D3=D4=\{1,2,3,4\}$ ,
  3.  $C = \{(C1,R1),(C2,R2),(C3,R3),(C4,R4)\}$  l'ensemble des contraintes, telles que  $C1 = (X1, X4)$ ,  $C2 = (X1, X2)$ ,  $C3 = (X1, X3)$ ,  $C4 = (X4,X2)$ ,  $R1 = \{(1,3),(1,2),(3,2)\}$ ,  $R2=\{(1,3),(2,2),(3,3)\}$ ,  $R3=\{(1,1),(2,2),(3,3)\}$ ,  $R4=\{(1,2),(2,3)\}$ .
 Représentez ce CSP sous la forme d'un graphe et donnez toutes les solutions possibles.

**Exercice 2 : Algorithme A\* (5 points)**

Dans un jeu video, un personnage géré par l'IA du jeu doit se déplacer sur un terrain en 3D pour atteindre sa destination. Le terrain est représenté par un quadrillage (fig. 1) constitué de cases qui ralentissent plus ou moins le déplacement du personnage. Par exemple, des montagnes ou des forêts ralentissent le déplacement du personnage (valeur plus élevée), alors que des plaines ou des routes facilitent son déplacement (valeur des cases plus faible).

Sur la figure 1, la lettre 'D' (respectivement 'A') représente le point de départ (respectivement d'arrivée) du personnage ; les cases infranchissables contiennent la lettre 'X'. Les nombres des autres cases indiquent le nombre d'unité de temps nécessaire pour les traverser. Sur la figure 1, le chemin représenté a une durée de :  $4+5+7+5+5+3=29$  unités de temps. Le personnage ne se déplace qu'horizontalement ou verticalement.

<b>D</b>	4	5	7
1	0	2	5
2	<b>X</b>	<b>X</b>	5
3	6	<b>A</b> ←	3

Fig. 1. Exemple de terrain 4x4 avec un chemin de valeur 15

Pour résoudre ce problème, on envisage d'utiliser l'algorithme A\*.

- 1) Indiquer pourquoi l'algorithme A\* est approprié pour résoudre ce problème ?
- 2) Donner une représentation possible d'un état du problème et définir un système de production (fonction de transition) permettant de passer d'un état à un autre.
- 3) On propose d'utiliser l'heuristique  $h1$  suivante :  $h1(s) = distMan(s,A)$   
où  $distMan$  représente la distance de Manhattan, donnée par :

$$distMan(M,N) = |X_N - X_M| + |Y_N - Y_M|$$

Peut-on utiliser l'heuristique  $h1$  pour le problème posé ? Expliquer pourquoi (en répondant éventuellement à la question : "Est-ce que cette heuristique est admissible ?").

- 4) Appliquez A\* pour résoudre l'instance du problème de la figure 1 avec l'heuristique  $h1$ , en détaillant pour chaque itération les valeurs des fonctions  $g$  et  $h$  ainsi que l'état suivant choisi, en utilisant ce type de tableau par exemple :

Itération	Liste <i>Open</i>	Liste <i>Closed</i>	Noeud N extrait de la liste <i>Open</i>	Valeur $g$ du noeud N	Valeur $h$ du noeud N	Valeur $f$ du noeud N
1						
...						

### Exercice 3 : Algorithme Alpha-Bêta (5 points)

Nous voudrions utiliser l'algorithme Alpha-Bêta pour savoir qui est le vainqueur du jeu de Nim suivant. On part d'une pile de  $n$  jetons. A tour de rôle, un des deux joueurs doit diviser en deux piles non vides et de tailles différentes une des piles devant lui. A chaque tour de jeu une nouvelle pile est donc créée. Par exemple, à partir d'une configuration de jeu comportant 3 piles respectivement composées de 2, 3 et 2 jetons, le seul coup jouable est de diviser la seconde pile en 2 et 1 jeton. Le joueur qui ne peut plus jouer a perdu. L'IA sera le second à jouer et elle est supposée minimiser ses gains, tandis que l'adversaire les maximise. Un état devra être représenté par une liste de piles triées selon leur taille. Seules les piles de taille supérieure à 2 doivent apparaître. Exemple : si un état contient 4 piles de taille respective 4,3,1,2, l'état correspondant est (4,3).

#### Questions :

1. Appliquez l'algorithme Alpha-Bêta en considérant une pile de départ de  $n=8$  jetons. Précisez pour chaque nœud de l'arbre de jeu ses bornes finales [alpha; bêta], indiquez avec une croix les coups qui seront élagués par l'algorithme et donnez enfin le premier coup qu'effectuera l'IA.
2. Qui gagne ?

### Exercice 4 : Prolog (5 points)

Soit un ensemble d'objets caractérisés par leur couleur (rouge (r), vert (v), bleu (b), jaune (j) ou orange (o)), leur forme (carré (ca) ou circulaire (ci)) et leur taille (longueur d'un côté ou rayon).

#### Questions :

1. Ecrivez le prédicat **objet( +L )** qui vérifie que L est un objet décrit par trois caractéristiques de couleur, forme et taille. Par exemple : `objet( [r,'ca',5] )` renvoie vrai, `objet( [4,'ca',5] )` renvoie faux (4 n'est pas une couleur valide) et `objet( [4,8] )` renvoie également faux.
2. Ecrivez le prédicat **objectSubset( +L,+O,?R )** qui est satisfait si R est l'ensemble d'objets extraits de L ne contenant que des objets de caractéristiques O. Dans O, une caractéristique définie à \* indique qu'elle n'est pas prise en compte dans la sélection des objets. Par exemple, si une taille de \* est indiquée dans O, les objets de toute taille respectant les deux autres critères seront présents dans R. Par exemple : `objectSubset([[r,'ca',5],[b,'ci',10]],[*,'*',5])` renvoie le premier objet, `objectSubset([[r,'ca',5],[b,'ci',10]],[*,'ci','*'])` renvoie le second objet.