

Coefficient : 40%.

Aucun document autorisé.

MERCI d'utiliser une copie différente pour chaque partie.

Remarques et conseils :

- Lisez **attentivement** chaque question.
- Indiquer clairement sur votre copie le numéro de l'exercice avant d'y répondre.
- Expliquez autant que possible vos choix.
- Le barème défini ci-après est susceptible d'être modifié.

Partie I : Questions diverses (12 points)

Exercice 1 : Questions de cours (4 points)

1. Le principe de résolution permet de déterminer si un ensemble de clauses est contradictoire ou non. Pour être appliqué sur des formules exprimées dans la logique des prédicats du premier ordre, le principe de résolution nécessite donc de transformer ces formules. Rappeler les trois étapes qui permettent de passer d'une formule quelconque à un ensemble de clauses.
2. Démontrer à l'aide du principe de résolution la validité ou la contradiction de la formule suivante :

$$\exists x \forall y \neg(S(x) \Leftrightarrow S(y)) \wedge \forall x S(x)$$

3. Quelle structure de données est manipulée par l'algorithme A* ? Que permet-il de déterminer ? Quelle est la particularité de A* et comment la définiriez-vous ?

Exercice 2 : Fonction LISP et prédicats PROLOG (4 points)

1. Que réalise la fonction LISP suivante. Donnez le résultat après son application sur un exemple pertinent.

```
(defun mystere (L)
  (cond
    ((atom L) L)
    ((member (car L) (cdr L)) (mystere (cdr L)))
    (t (cons (car L) (mystere (cdr L))))))
```

2. Ecrivez le prédicat PROLOG *phrase(L)* qui permet de reconnaître les phrases spécifiées par la grammaire suivante :

phrase → *groupeNominal groupeVerbal*
groupeVerbal → *auxiliaireEtre participe*

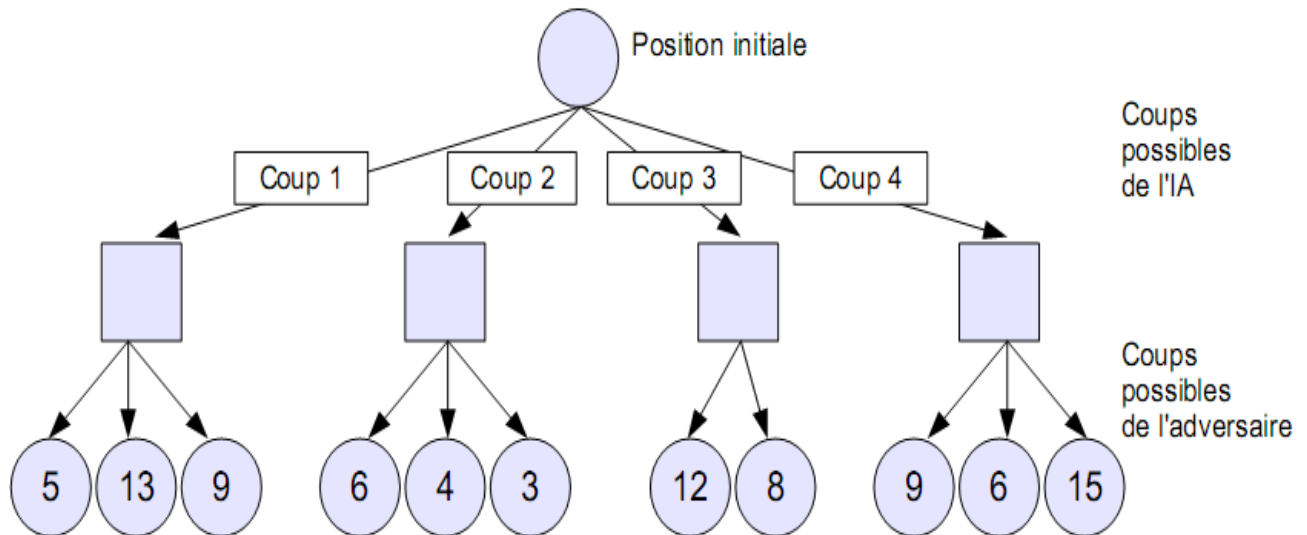
<i>groupeNominal</i>	→ <i>prenom</i> <i>prenom et prenom</i>
<i>auxiliaireEtre</i>	→ <i>est</i> <i>sont</i>
<i>participe</i>	→ <i>parti</i> <i>partis</i> <i>partie</i> <i>parties</i>
<i>prenom</i>	→ <i>'Marc'</i> <i>'Hervé'</i> <i>'Marie'</i> <i>'Emma'</i>

phrase représente l'axiome et les termes soulignés sont des symboles terminaux. Les phrases pouvant être reconnues doivent respecter l'accord en genre et en nombre. « Paul est parti », « Marie est partie », « Paul et Hervé sont partis », « Paul et Marie sont partis » sont des phrases correctes, « Paul et Hervé est parties » ne l'est pas.

Exercice 3 : Algorithmes Minimax et Alpha-Bétâ (4 points)

Ces algorithmes, Minimax et Alpha-Bétâ, sont largement utilisés pour déterminer le meilleur coup qu'une intelligence artificielle (joueur cybernétique) doit réaliser lorsqu'elle est confrontée à un autre joueur dans le cadre d'un jeu à somme nulle.

L'arbre ci-dessous représente l'arbre de recherche d'un tel jeu à deux joueurs. L'évaluation de chaque position pouvant être atteinte à un niveau de profondeur 2 à partir de la position initiale est indiquée sur les feuilles de l'arbre.



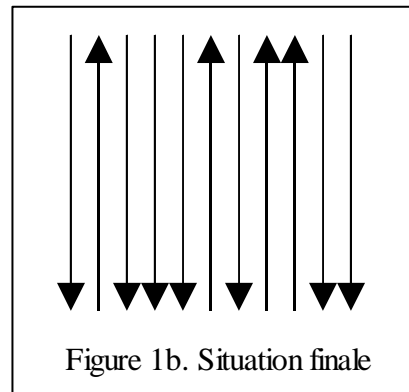
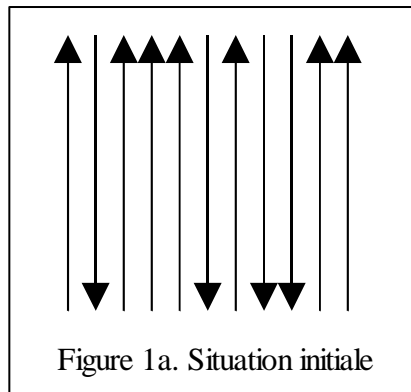
Questions :

1. Précisez quel est l'objectif de l'IA et quel est l'objectif de son adversaire.
2. En utilisant l'algorithme Minimax, précisez pour chaque noeud son évaluation et indiquez le coup qu'effectuera l'IA.
3. En utilisant l'algorithme Alpha-Bétâ, précisez pour chaque noeud ses bornes finales [alpha; bétâ]. Indiquez également avec une croix les coups qui seront élagués par l'algorithme et donnez enfin le coup qu'effectuera l'IA.
4. Quel critère dans la construction de l'arbre de recherche permettrait d'améliorer l'efficacité de l'élagage de Alpha-Bétâ ? Si ce critère est appliqué, combien de noeuds dans l'arbre de recherche ci-dessus pourraient alors être élagués ?

Partie II : Résolution de problème et LISP (8 points)

Exercice 4 : Problème des flèches (8 points)

Dans le problème des flèches, n flèches sont positionnées verticalement et orientées soit vers le haut soit vers le bas. On souhaite inverser l'orientation de toutes les flèches, c'est-à-dire que celles qui étaient orientées vers le haut soient orientées vers le bas et que celles qui étaient orientées vers le bas, pointent vers le haut (voir figures 1a et 1b ci-dessous).

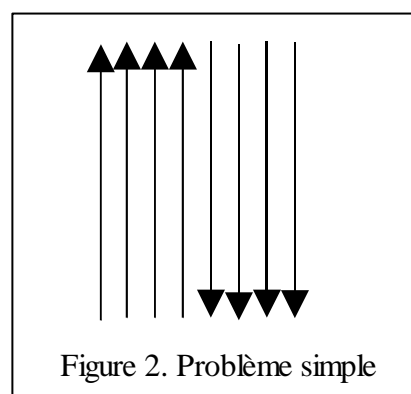


On peut faire évoluer la situation dans les deux cas suivants :

- en retournant un paquet de trois flèches adjacentes ayant la même orientation (haut ou bas),
- en retournant deux flèches adjacentes ayant des orientations inverses (une orientée en haut et l'autre orientée en bas ou inversement).

Questions :

- 1) Donner une représentation d'un état de ce problème.
- 2) Formaliser les règles de production qui permettent de passer d'un état à un autre et donc de construire l'espace d'états. Pour chaque règle, préciser ses conditions d'application ainsi que ses effets.
- 3) On envisage de résoudre ce problème en utilisant A^* . Rappeler la définition des fonctions g et h dans le contexte de A^* .
- 4) Définissez les fonctions g et h pour ce problème.
- 5) Résoudre le problème des 8 flèches de la figure 2 à l'aide des fonctions g et h de la question 4. Détailler à chaque étape, les valeurs des fonctions g et h .



- 6) Définissez une représentation d'un état du système en LISP.
- 7) Ecrivez la fonction LISP qui calcule la valeur de la fonction h pour un état du système.