

INTELLIGENCE ARTIFICIELLE

Correction TD3

2^{ème} année ING Informatique
Groupes A, B, C et D

ALGORITHMES DE JEUX

Exercice 1

Quels sont les jeux où l'algorithme MinMax peut être efficace ?

- Poker
- Pierre / feuille / ciseaux
- Echecs
- Belotte
- Dames
- Reversi
- Tic-Tac-Toe
- Go

Correction

Minimax est un algorithme s'applique efficacement à la théorie des jeux pour les jeux à deux joueurs à somme nulle et information complète.

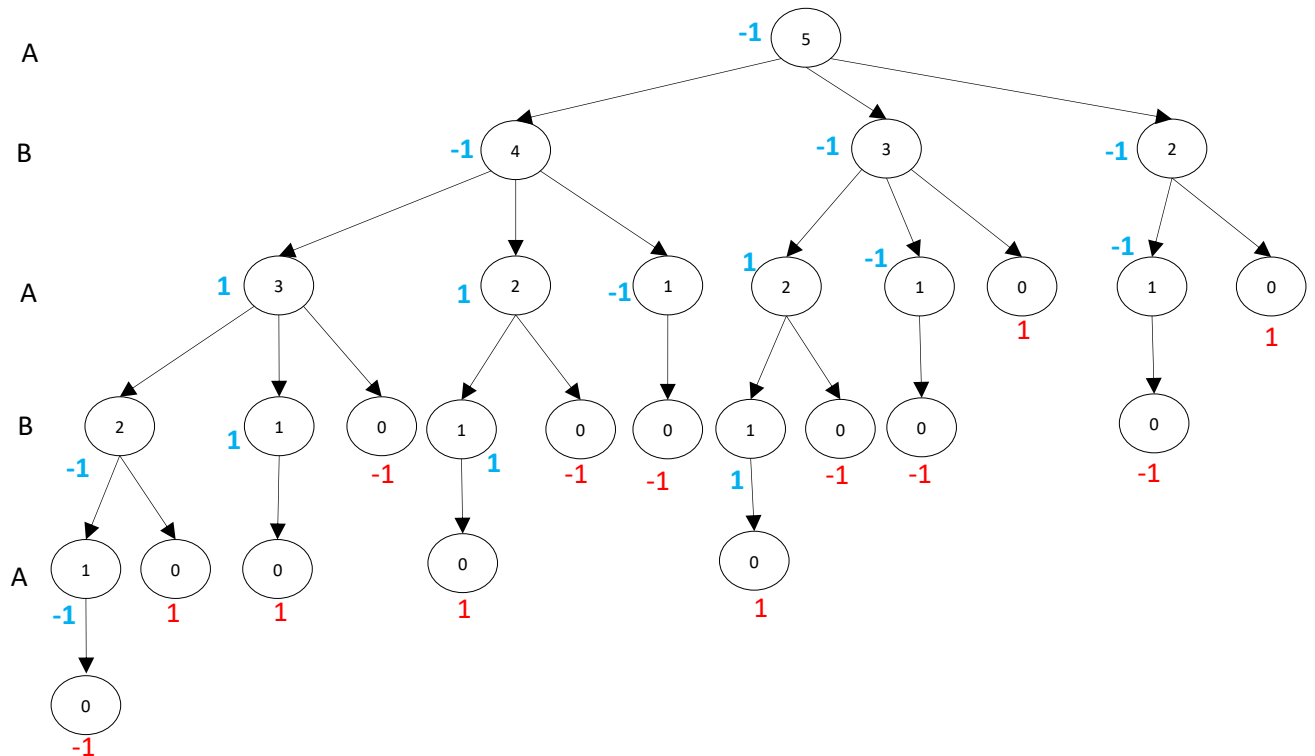
- Poker
- **Pierre / feuille / ciseaux**
- **Echecs**
- Belotte
- **Dames**
- **Reversi (Othello)**
- **Tic-Tac-Toe**
- **Go**

Exercice 2

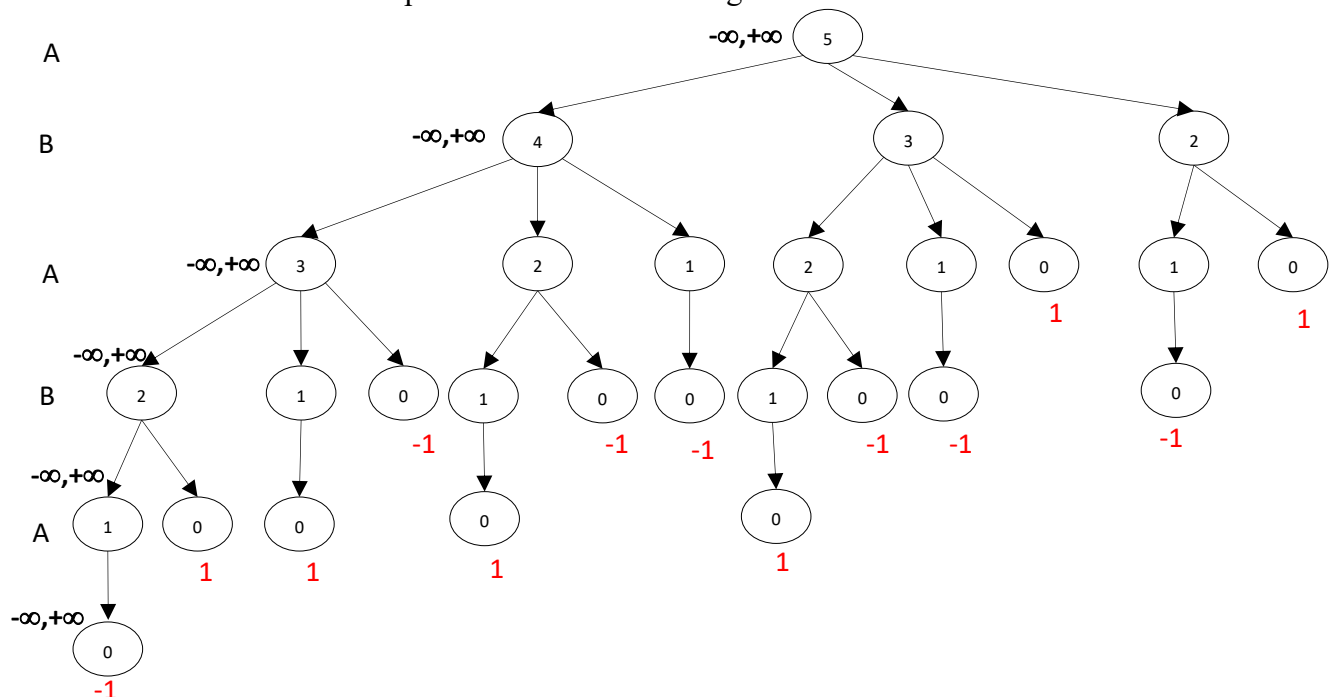
Les jeux de Nim sont des jeux de duel à somme nulle. Les joueurs disposent d'un tas d'objets (graines, allumettes ...) et chaque joueur enlève 1, 2 ou 3 objets à tour de rôle. Le perdant est celui qui prend le dernier objet. Dans cet exercice nous allons considérer un jeu à 5 objets.

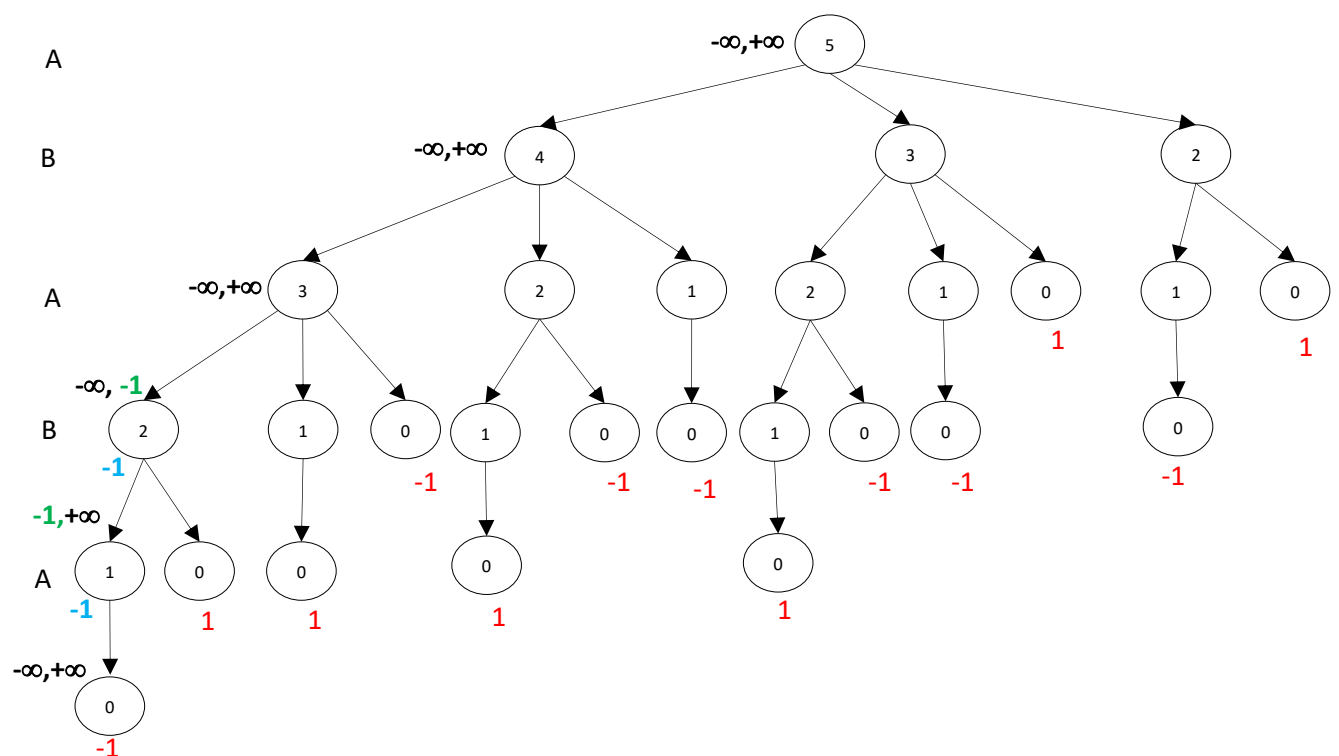
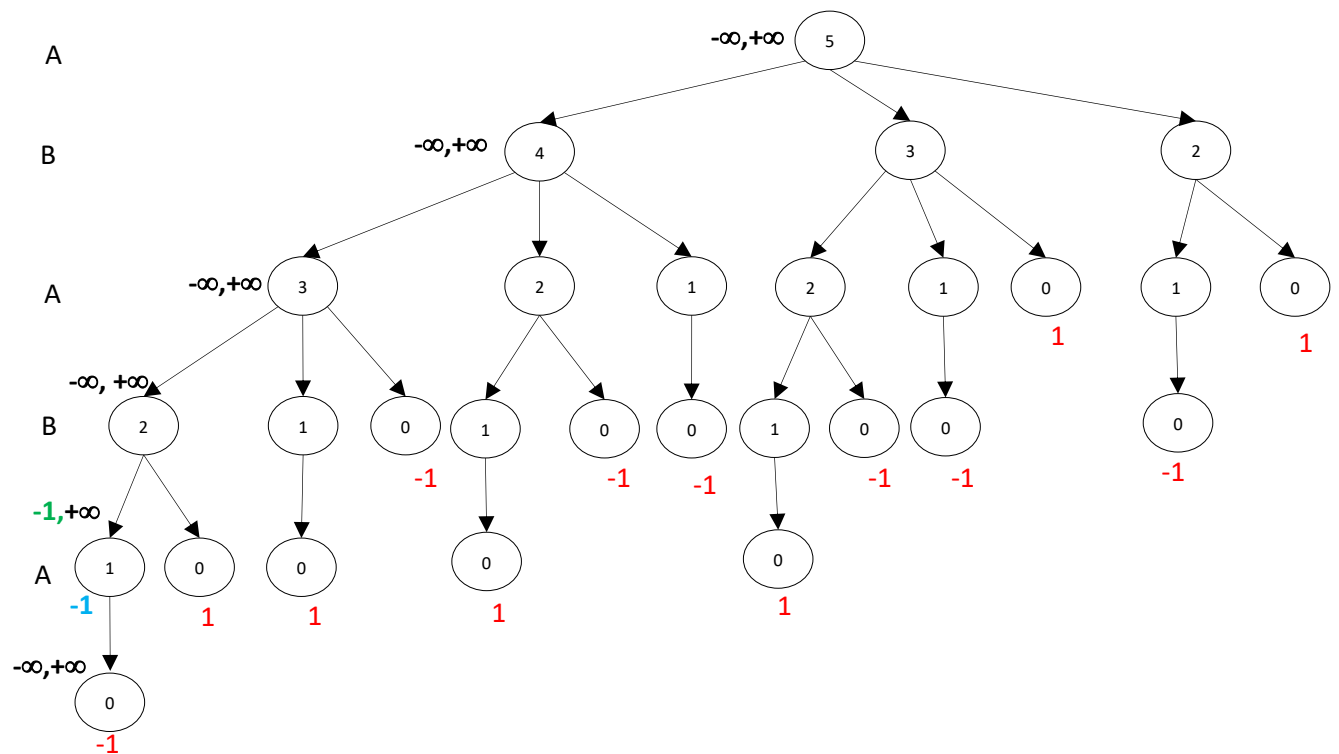
1. Dessiner l'arbre de recherche complet du jeu.
2. Appliquer l'algorithme MinMax.
3. Est-on certain de gagner si l'on joue en premier ? En second ?
4. Appliquer l'algorithme MinMax avec élagage Alpha-Beta.

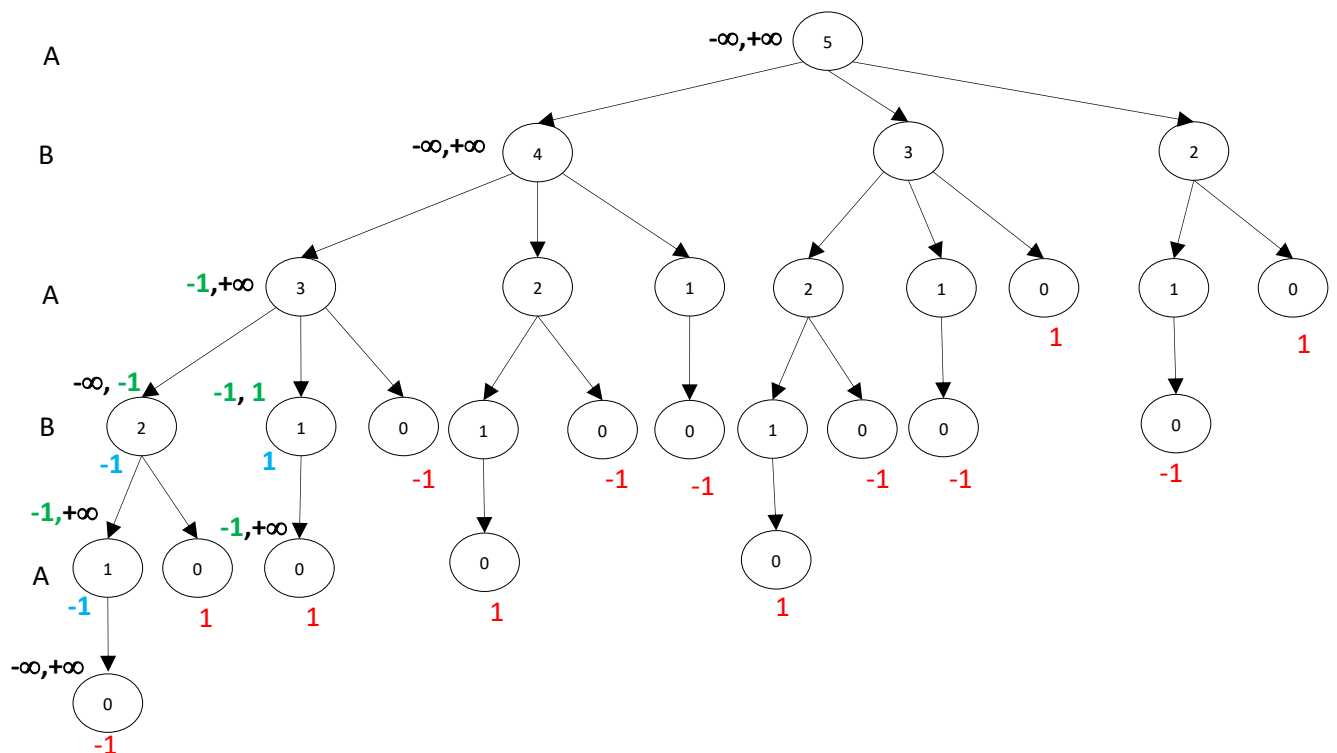
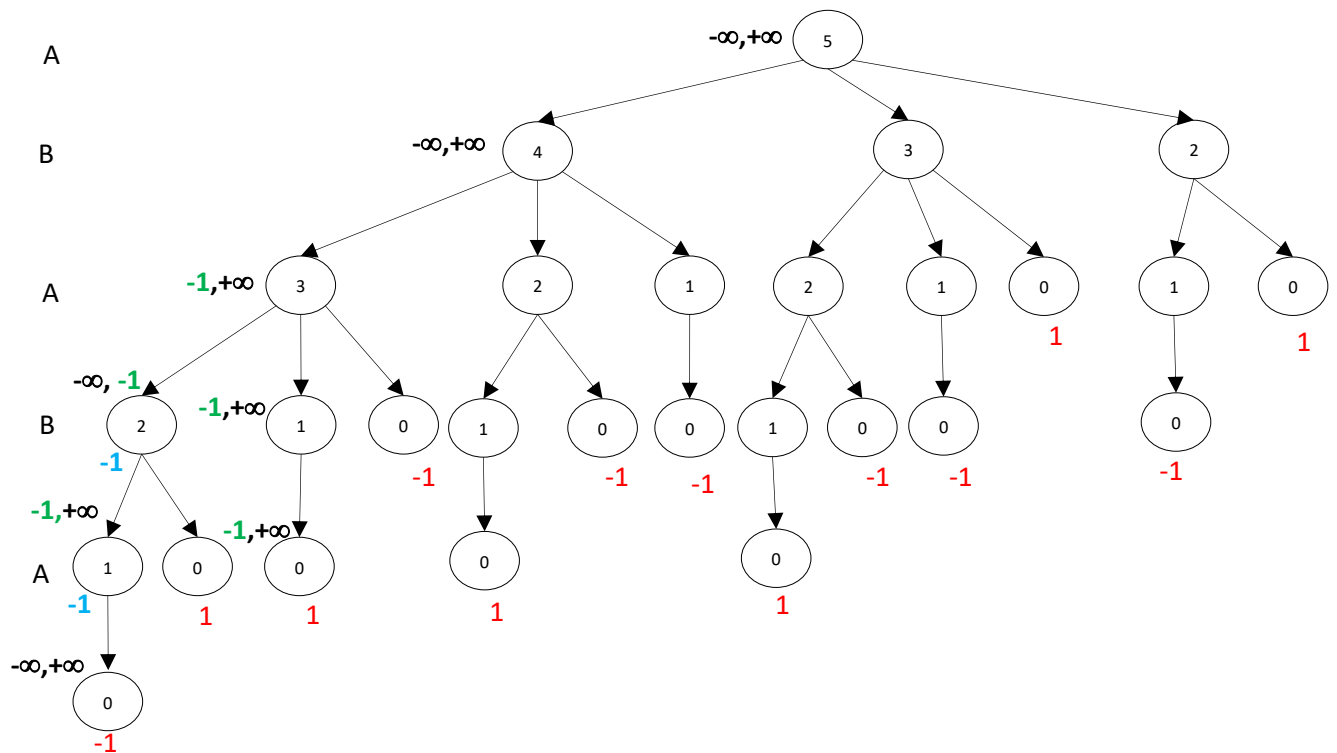
Minimax permet de retourner l'action optimale offrant la meilleure valeur pour chaque joueur, en remontant l'appel récursif à partir des feuilles et jusqu'à la racine :

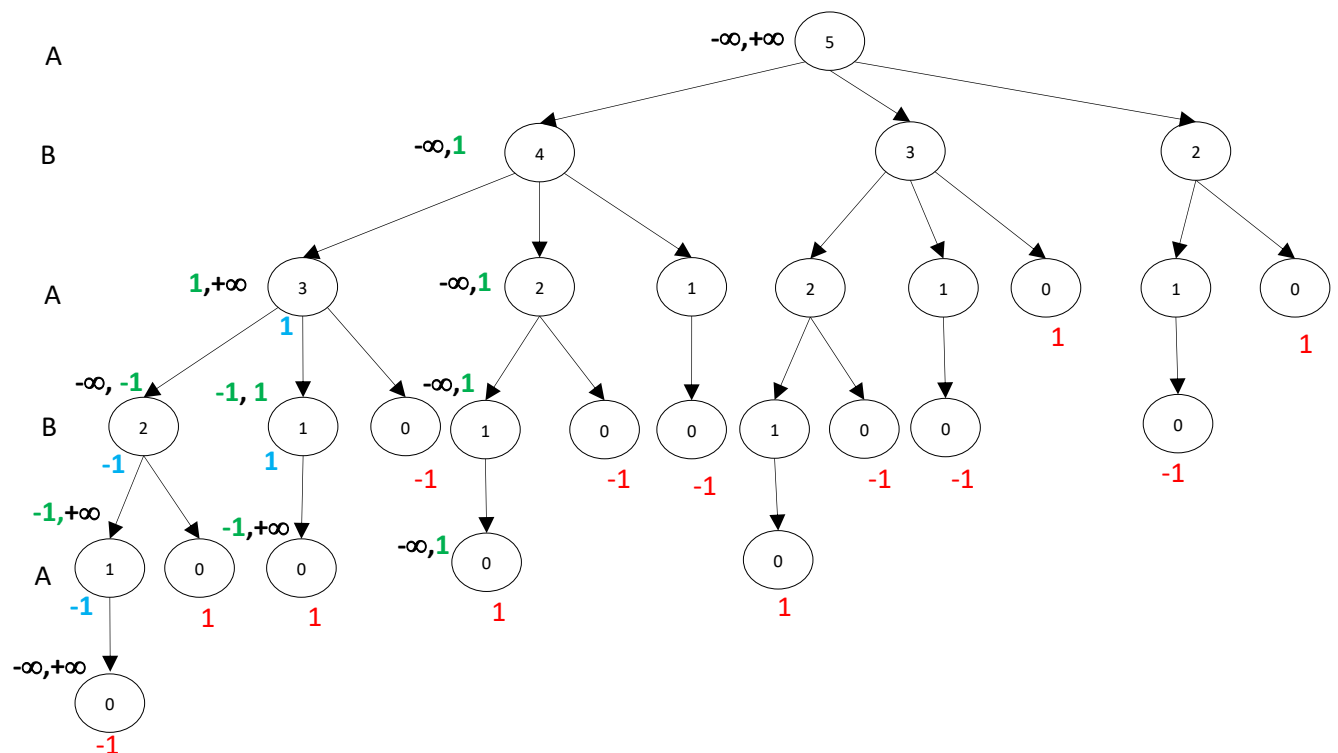
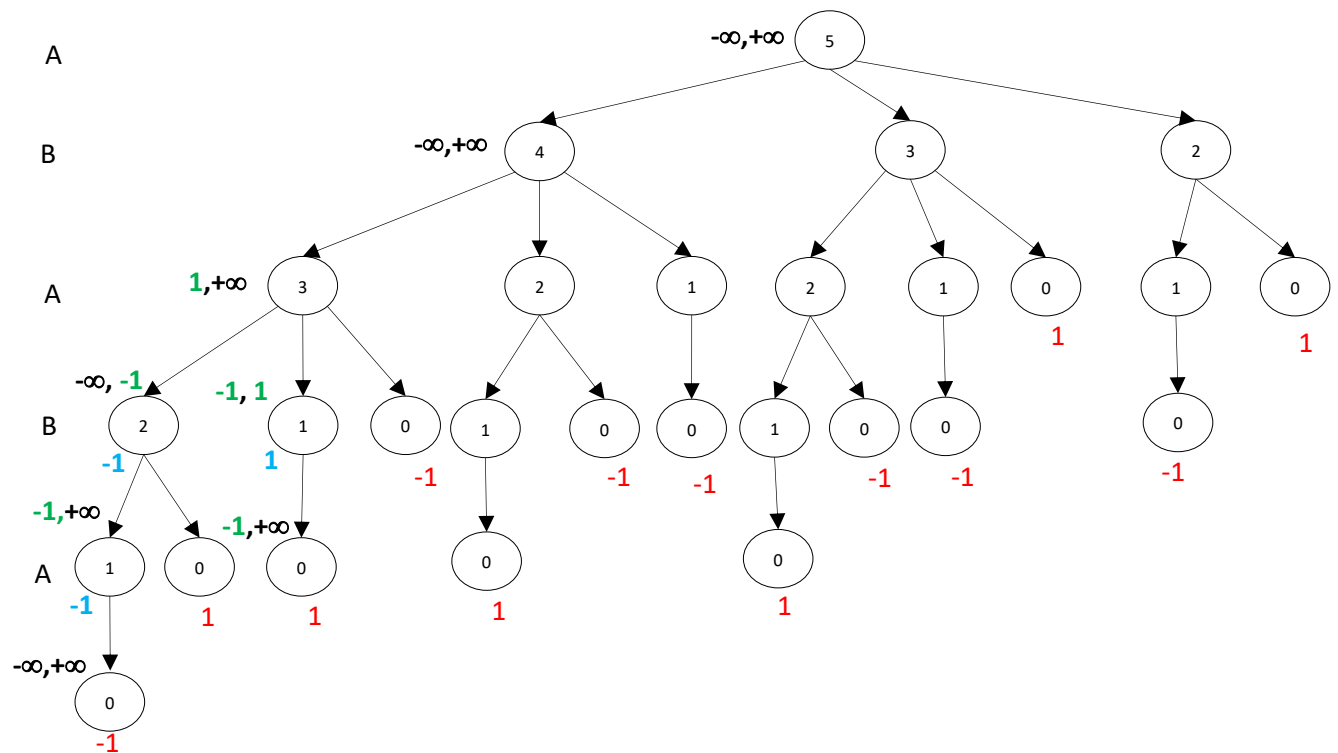


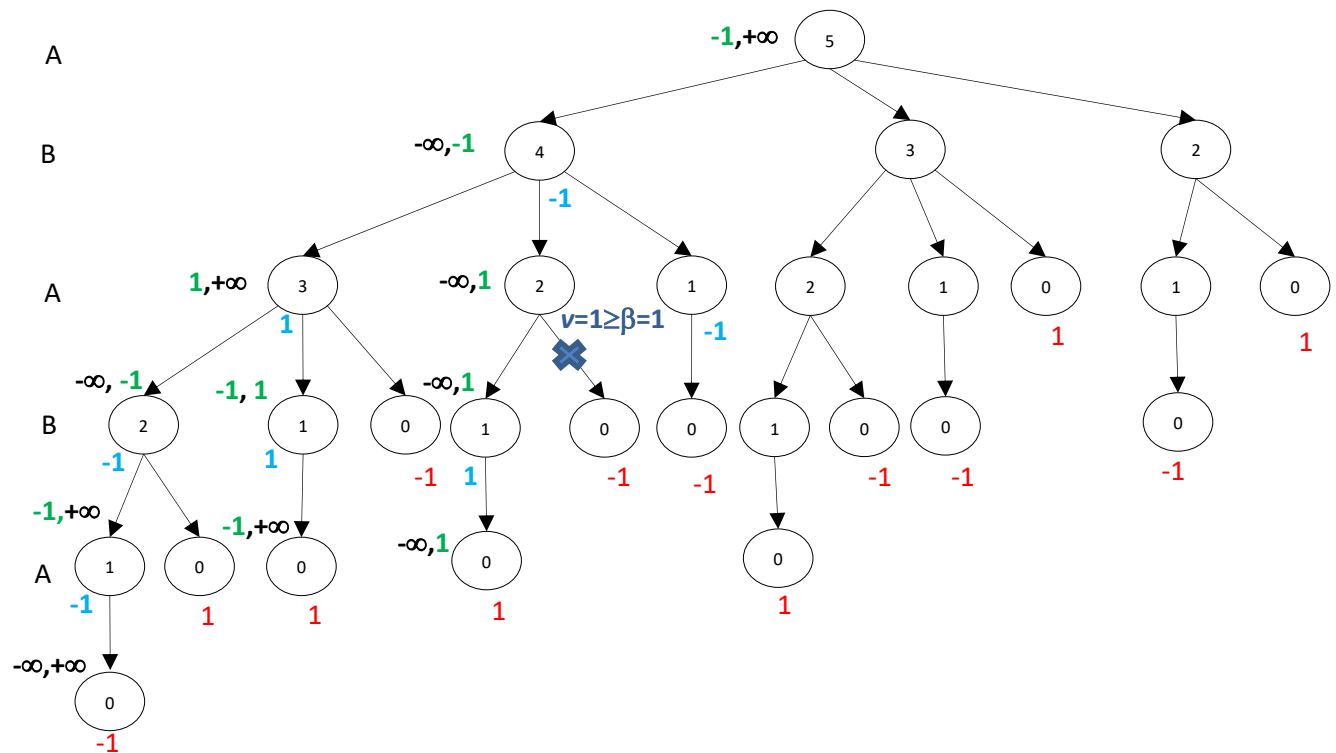
3. Celui qui joue en premier perd à tous les coups, bien sûr B doit jouer optimalement. A, qui joue en premier a le choix entre les états : 4, 3 et 2 :
 - a. S'il joue 4, B jouera 1, ensuite A jouera 0 : B gagne
 - b. S'il joue 3, B jouera 1, ensuite A jouera 0 : B gagne
 - c. S'il joue 2, B jouera 1, ensuite A jouera 0 : B gagne
4. Ci-bas les différentes étapes de l'exécution de l'algorithme :

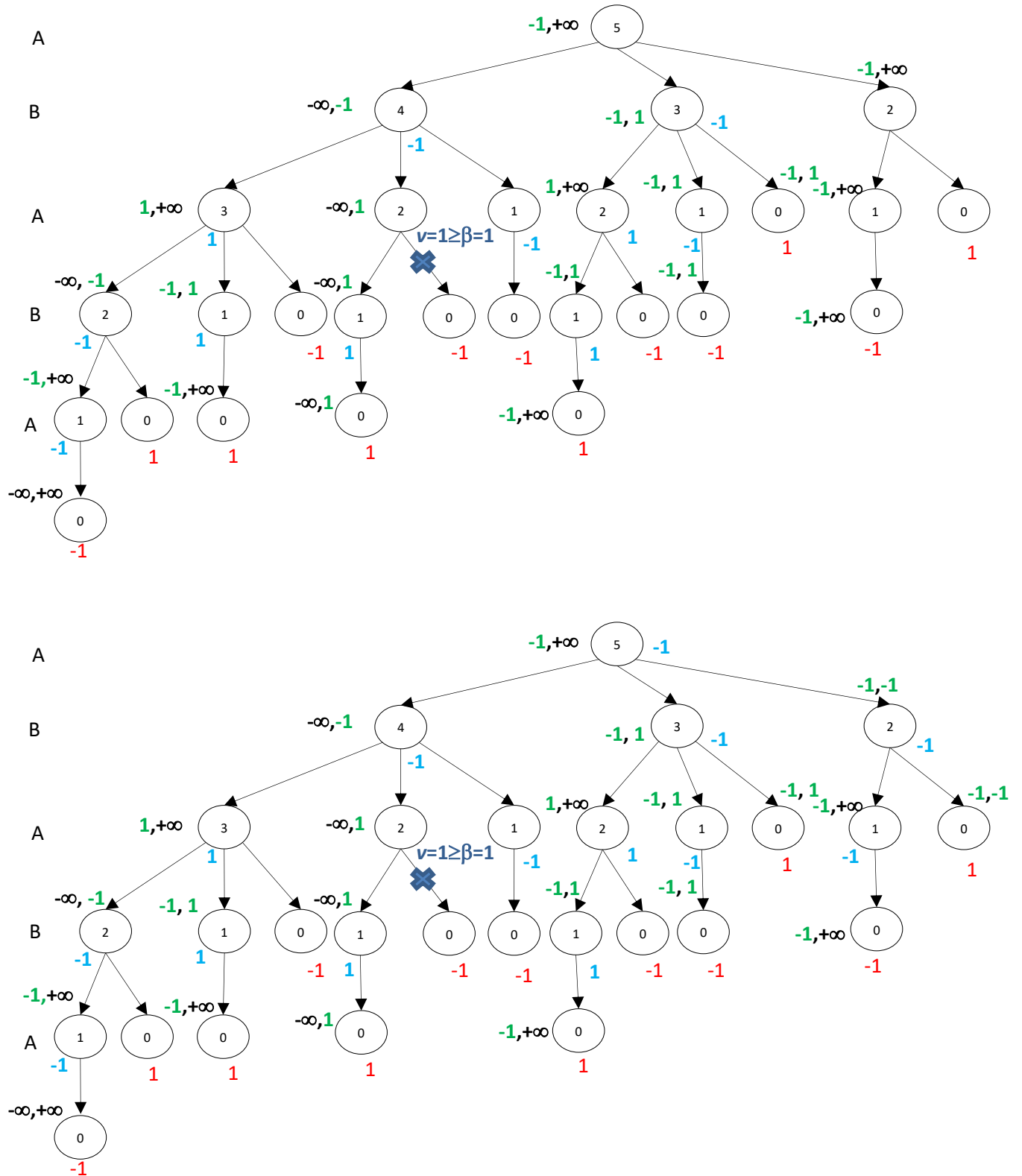






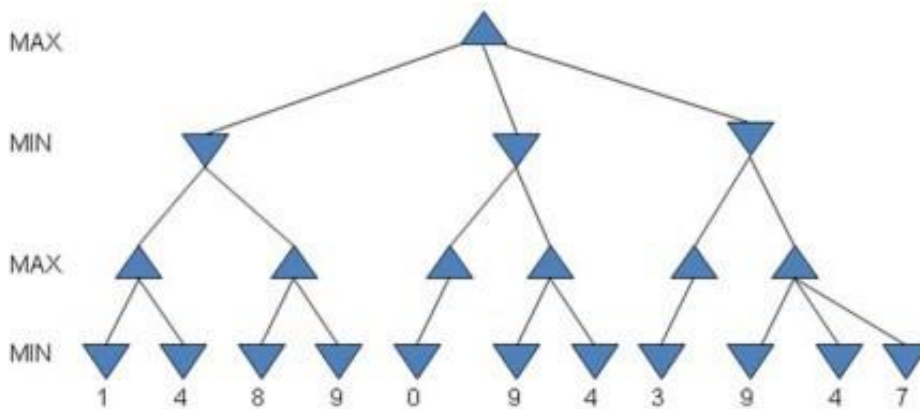






Exercice 3

1. Soit l'espace d'états suivant modélisant les actions de deux joueurs (MAX et MIN). Les feuilles correspondent aux états terminaux du jeu. Les valeurs des états terminaux sont indiquées en bas de chaque état. Dessinez la partie de l'espace d'états qui serait explorée par l'algorithme alpha-beta pruning, en supposant qu'il explore l'espace d'états de la gauche vers la droite. Dessinez seulement les états explorés et les transitions correspondantes. Indiquez, à côté de chaque état exploré, la valeur correspondante à la terminaison de l'algorithme.



Correction :

$A \leq \beta = 4$ (A ne sera jamais plus grand que 4)

B est β -coupé puisque $8 > \beta = 4$

$C \geq \alpha = 4$ (C ne sera jamais plus petit que 4)

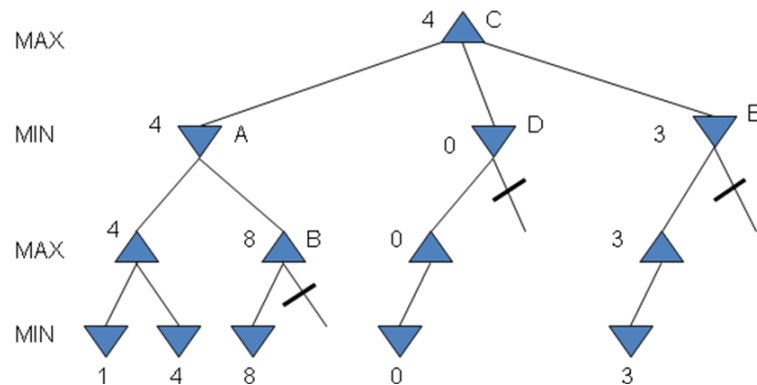
D est α -coupé puisque $0 < \alpha = 4$

E est α -coupé puisque $3 < \alpha = 4$

C a la valeur 4.

Rappel :

- α : meilleure (plus grande) valeur de MAX jusqu'ici; elle ne décroît jamais; MAX ne considère jamais les nœuds (en bas de lui – successeurs) ayant des valeurs plus petites que α .
- β : meilleure (plus petite) valeur de MIN jusqu'ici; elle ne croît jamais; MIN ne considère jamais les nœuds (en bas de lui – successeurs) ayant des valeurs plus grandes que β .



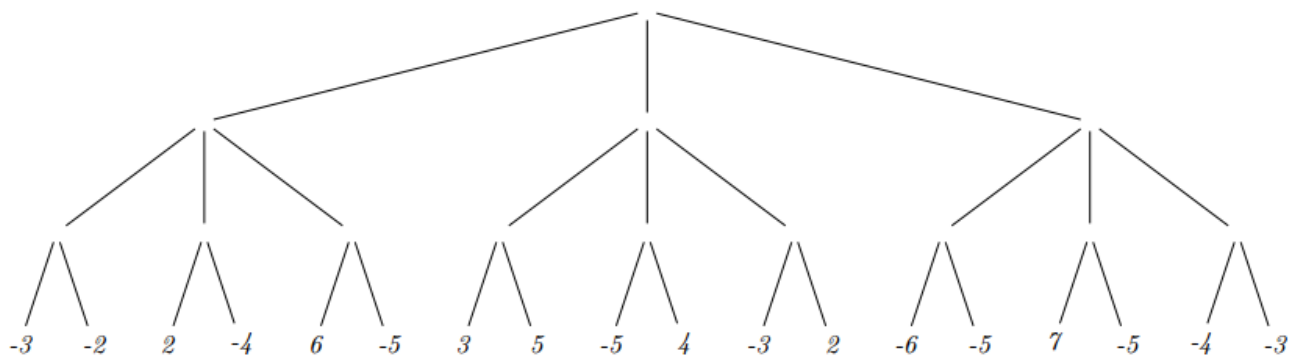
2. Supposons que l'algorithme alpha-beta pruning explore le même espace d'états, mais cette fois-ci de la droite vers la gauche. À la terminaison, l'état à la racine aura-t-il la même valeur que dans l'exploration précédente ? Répondez par oui ou non et ensuite expliquez clairement pourquoi.

Correction :

Oui. Parce que *alpha-beta pruning* donne toujours le même résultat que minimax; par contre, les états explorés ou coupés ne seront pas forcément les mêmes selon qu'on explore l'espace de gauche vers la droite ou de droite vers la gauche.

Exercice 4

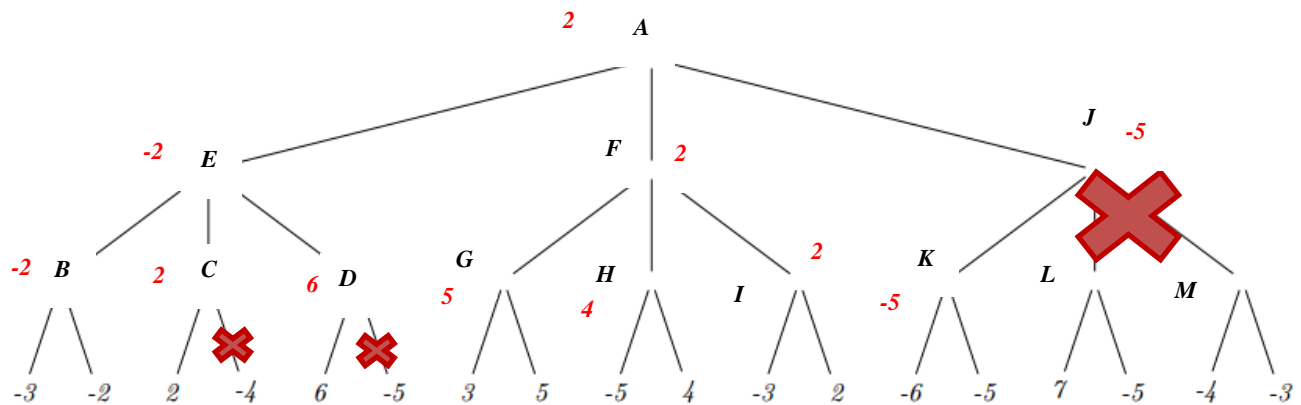
On considère un jeu à deux joueurs dont l'arbre de jeu est donné par le schéma ci-dessous.



1. Appliquer l'algorithme Alpha-Beta à ce jeu

Correction :

On suppose que Max commence.



C est un nœud Max, donc il va chercher à maximiser son gain. Son premier fils a la valeur 2, alors que son domaine est $[-\infty, -2]$ (domaine mis à jour après l'exploration de B : comme E est un nœud Min et que son premier fils B a la valeur -2). Comme v du premier fils de C a la valeur $2 \geq \beta = -2$, le reste du sous arbre ne sera pas exploré.

De même pour le nœud D, $v=6 \geq \beta=-2$, donc pas d'exploration du reste de l'arbre.

Après l'exploration du fils E, le domaine de A s'étend à $[-2, +\infty]$.

Après l'exploration du fils de K de J, v (au niveau de J) prend la valeur -5 et comme $v=-5 \leq \alpha=2$, v sera retourné et les autres fils de J ne seront pas explorés. Ceci est normal puisque, le nœud J, comme nœud MIN, aura une valeur maximale de -5 et donc ce nœud ne sera jamais choisi par A puisqu'il a une meilleure option (le nœud F avec une valeur de 2).