L’algorithme du Minimax

Introduction

Contexte

L’algorithme du Minimax est un algorithme permettant de résoudre les jeux de stratégie opposant deux joueurs tels que le jeu d'échecs ou le jeu de go. Il fait appel à des notions fondamentales en informatique telles que les arbres ou la récursivité. C’est également un exemple d’algorithme de la théorie des jeux ou de l'intelligence artificielle.
C’est un point d’entrée pour comprendre comment un ordinateur “apprend” ou "réfléchi". Il est susceptible d’intéresser un élève de lycée car son application permet de gagner à coup sûr à un certain nombre de jeux.

Descriptif

Le but de cette activité est de présenter l’algorithme du Minimax qui permet de résoudre un jeu avec un ordinateur. Cet algorithme fait appel à des notions de modélisation en informatique, en particulier il utilise un graphe pour représenter l’ensemble de positions possibles au cours d’une partie et utilise une fonction pour évaluer chacune des positions..
Ainsi lors de l’activité

- on introduira la structure de donnée de l’arbre en informatique
- on travaillera sur la représentation des états du jeu
- on travaillera autour de la fonction d’évaluation
- enfin on présentera l’algorithme du Minimax

Informations pratiques

Niveau : Lycée
Matériel : 2 boites d’allumette (8 allumettes par élève), un tableau, fiches élèves
Durée: 1h28
Compétences requises:

- Notions sur les fonctions

Déroulement

- <u>Présentation du jeu et découverte du jeu 10’</u>
  Cette première partie consiste à regrouper les élèves par binôme et les faire jouer au jeu de NIM.
- <u>Découverte des arbres 8’</u>
  Cette partie s’appuie sur la fiche d’activité partie 1. Elle doit sembler aux élèves assez facile. On introduit de façon intuitive les arbres et le vocabulaire associé. On

laissera 5’ aux élèves pour répondre aux questions et 3’ pour la correction au tableau.

- La représentation d’une partie 10’
  Cette partie s’appuie sur la fiche d’activité partie 2. On introduit une façon de représenter formellement une petite partie du jeu de NIM (avec seulement 5 allumettes en début de partie). On laissera 6’ aux élèves pour répondre aux questions et 4’ pour la correction au tableau.

- L’arbre des possibles 15’
  Cette partie s’appuie sur la fiche d’activité partie 3, elle consiste à définir l’arbre de jeu pour une étape du jeu de NIM (avec seulement 6 allumettes en début de partie). Elle doit aussi permettre aux élèves de comprendre ce que signifie “position gagnante” en s’appuyant sur le cas d'exemple du jeu de NIM à 6 allumettes (le joueur qui commence est dans une position gagnante). On laissera 10’ aux élèves pour répondre aux questions et 5’ pour la correction au tableau.

- Les positions gagnantes 15’
  Cette partie s’appuie sur la fiche d’activité partie 5, elle doit permettre aux élèves de comprendre comment on peut déterminer récursivement si un état est gagnant ou perdant. On retravaillera le cas d'exemple du jeu de NIM à 6 allumettes de la partie précédente afin d’associer pour chaque état du jeu un état “gagnant” ou “perdant”. On laissera 10’ aux élèves pour répondre aux questions et 5’ pour la correction au tableau.

- La fonction d’évaluation 15’
  Cette partie s’appuie sur la fiche d’activité partie 5, elle consiste à définir une fonction qui associe à un état du jeu une valeur indiquant pour qui la position est avantageuse. Jusqu’alors on utilisait des couleurs pour encoder les états “gagnant” et “perdant” car cela est plus intuitif mais le but de l’activité étant d’introduire l’algorithme du Minimax on définit un fonction à valeur dans les entiers. On laissera 10’ aux élèves pour répondre aux questions et 5’ pour la correction au tableau.

- L’algorithme du Minimax 15’
  Cette partie s’appuie sur la fiche d’activité partie 6, elle donne l’algorithme du Minimax (en pseudo code) et le fait appliquer aux élèves sur un exemple simple. Elle amène également à réfléchir autour de l’algorithme : pourquoi fournit il une solution pour résoudre un jeu ? On laissera 10’ aux élèves pour répondre aux questions et 5’ pour la correction au tableau.

Le lien avec l’informatique

Cette activité permet de découvrir des notions fondamentales en informatique telles que les arbres ou la récursivité. Elle fait également travailler des notions d'algorithmie et permet d’utiliser la notion de fonction dans le cadre de problèmes informatiques. Enfin elle peut consister en une introduction au domaine de l’intelligence artificielle et de de la théorie de jeux.

Gagner au jeu de NIM grâce à l'algorithme du Minimax

Comment jouer au jeu de NIM ?

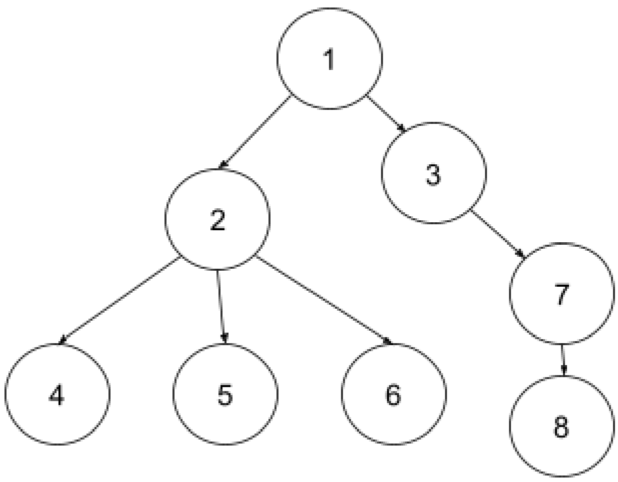
Les règles du jeu de Nim sont simples :

- répartir 16 allumettes en ligne et désigner un joueur pour commencer
- chaque joueur, à tour de rôle, doit prendre une, deux ou trois allumettes
- le joueur qui prend la dernière allumette a perdu.

1) Présentation des arbres

Un **arbre** est une structure de données très utilisée en informatique. Il est défini comme un ensemble de **nœuds**, et d'**arêtes**. On dit que le nœud A est le **père** du nœud B si il y a une arête allant du nœud A au nœud B.
La **racine** de l'arbre est le nœud qui n'a pas de père. Les **feuilles** sont les nœuds n'ayant pas de fils, un arbre peut être représenté graphiquement, les arêtes sont généralement représentées par des flèches et les nœuds par des cercles.

Voici une représentation d'un arbre en informatique



1. Surligne une arête de l’arbre
2. Colorie en bleu les fils du noeud 1
3. Colorie en rouge le père du noeud 8
4. Colorie en vert les feuilles de l’arbre
5. Colorie en violet la racine de l’arbre

2) Représenter une partie

On imagine qu’Alice et Bob jouent une partie du jeu de NIM. Pour un état donné de la partie, nous souhaitons représenter les informations suivantes:
- Qui doit jouer ?
- Combien y- a-t’il d’allumettes sur la table?

Ainsi on représente l'état où c’est au tour d’Alice de jouer et il reste 5 allumettes par le noeud suivant

1. Que représente le nœud sur la figure 2.1 ?
2. Pour représenter une partie, on utilise une séquence de nœuds. Par exemple la figure 2.2 représente la partie suivante:
    - Alice commence avec 5 allumettes et elle en retire 1.
    - Puis c’est au tour de Bob qui a 4 allumettes. Il en retire 1.
    - Enfin Alice retire 2 allumettes et Bob a perdu.

    En figure 2.3:
    - Combien y a-t-il d'allumettes au début de la partie ?
    - Qui a joué au 2ème tour ?
    - Qui a gagné ?
    - Combien Alice a retiré d’allumettes lors de son 2ème coup ?
    - Y-a-t’il un problème ? Si oui lequel ?

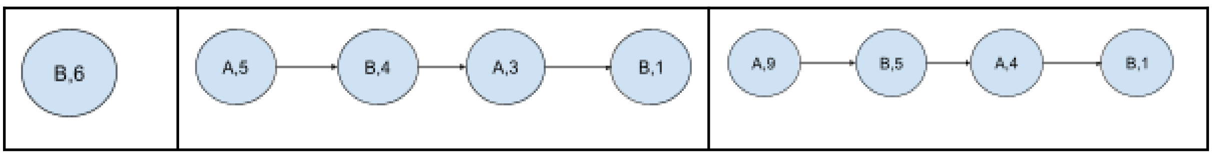


Figure 2.1	Figure 2.2	Figure 2.3

3) L'arbre des possibles

On cherche à représenter l'ensemble des positions possibles.

1. Complète l'arbre figure 3.1 pour qu'il représente l'ensemble des positions possibles après qu'Alice ait joué.
2. Joue au jeu de NIM avec 6 allumettes. Qui a l'avantage en début de partie ?
3. Que doit jouer Bob pour son premier coup ? A-t-il l'avantage ? Pourquoi ?
4. Complète l'arbre des coups possibles représenté en figure 3.2 en indiquant pour chaque nœud qui doit jouer et le nombre d'allumettes.

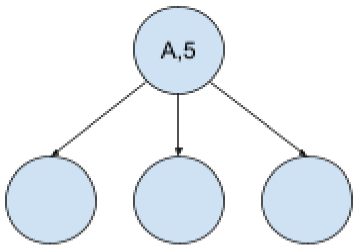


Figure 3.1

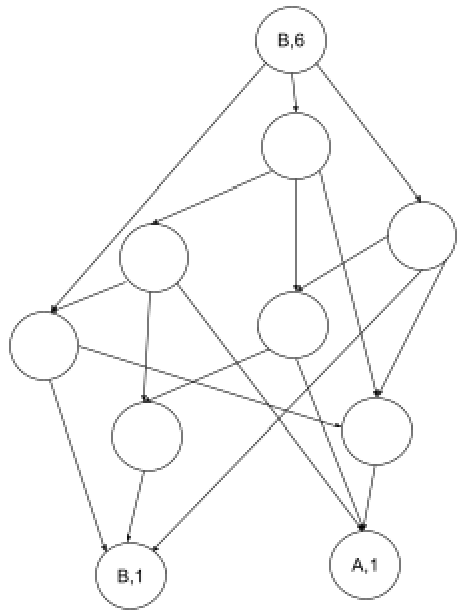


Figure 3.2

4) Déterminer les positions gagnantes

Dans la suite on décide de colorier l'arbre des possibles trouvé précédemment. On souhaite colorier en vert les nœuds où Bob est dans une position gagnante, c'est-à-dire les nœuds tels que, quelque soit les coups d'Alice, il peut s'assurer de gagner.

1. De quelle couleur colorierais tu les nœuds blancs des figures 4.1 à 4.4 ? Pourquoi ?
2. Sur la figure 3.2 de la section précédente, colorie en vert les feuilles où Bob gagne et en rouge les feuilles où Alice gagne.
3. Entoure les nœuds dont tous les enfants sont coloriés.
4. Sur la figure 3.2, pour les nœuds dont tous les enfants sont coloriés, colorie en vert ceux où Bob gagne et en rouge ceux où Alice gagne (en supposant que chacun d'eux va jouer le meilleur coup). Recommence la procédure précédente jusqu'à avoir colorier tous les nœuds.

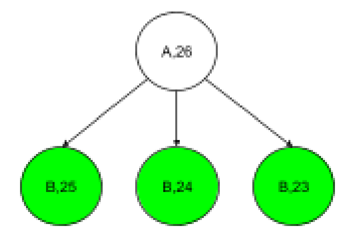


Figure 4.1

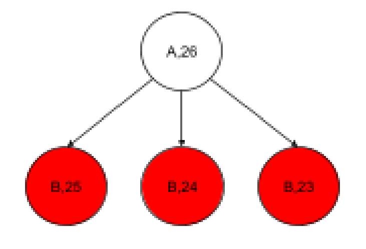


Figure 4.2

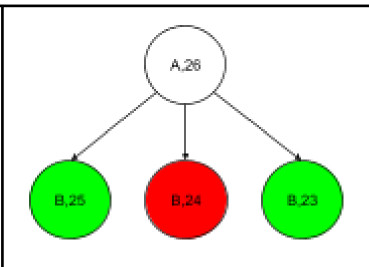


Figure 4.3

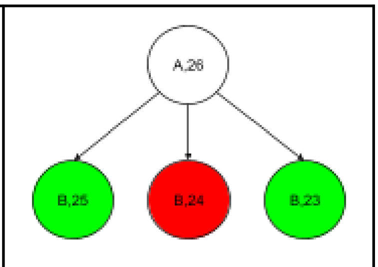
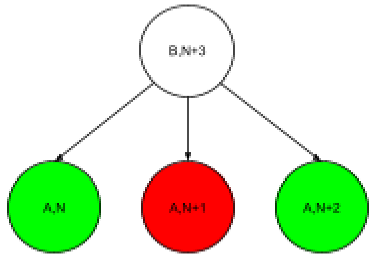


Figure 4.4

5) Une fonction d'évaluation

On décide de définir une fonction f, pour indiquer si une position est gagnante pour Alice ou pour Bob.
On dit que f(X,N) vaut 0 si la position est gagnante pour Alice et f(X, N) vaut 1 sinon (X pouvant être Alice ou Bob et N étant le nombre d'allumettes). 0 correspond donc à la couleur rouge et 1 à la couleur verte. On utilise des nombres plutôt que des couleurs car en informatique les langages de programmation utilisent des nombres plutôt que des couleurs.

1. Que valent f(A,1) et f(B,1) ?
2. En supposant que f(A,44)=1 colorie le nœud en figure 5.1.
3. Colorie le nœud blanc en figure 5.2. D'après ce schéma que valent f(A,N) et f(B,N+3)?
4. Peut on dire pour qui la position est gagnante s'il y a N+3 allumettes, que c'est à Bob de jouer et que l'on connaît f(A,N+2), f(A,N+1), f(A,N) ?
5. Complète le tableau 5.1. Pour chaque case, donne f(B,N+3) et f(A,N+3), connaissant f(X,N+2),f(X, N+1),f(X, N).

	
Figure 5.1	Figure 5.2

f(X, N+2),f(X, N+1),f(X, N)	0,0,0	1,0,0	0,1,0	0,0,1	1,1,0	1,0,1	0,1,1	1,1,1
f(A,N+3)								
f(B,N+3)								

Tableau 5.1

6) L'algorithme du Minimax

On donne ci-dessous l'algorithme du Minimax.

Algorithme **Minimax**(noeud):

Si le noeud (X,N) est une feuille :

si X = "Bob": f(X,N) prend la valeur 0

sinon f(X,N) prend la valeur 1

Sinon pour chaque fils f du noeud :

applique Minimax(f)

si X = "Alice": f(X,N) prend la valeur minimale des valeurs des nœuds enfants.

sinon: f(X,N) prend la valeur maximale des valeurs des nœuds enfants.

1. Applique l'algorithme à l'arbre représenté en figure 5.1. en partant de la racine. (Associe à chaque noeud un numéro : 0 ou 1)
2. Que fait l'algorithme du Minimax?
3. Pourquoi utilise t'on un maximum quand c'est au tour de Alice et un minimum quand c'est au tour de Bob?
4. Colorie les nœuds de l'arbre représenté en figure 6.1.
5. En supposant que Bob ait accès à l'arbre de la figure 6.1 que doit il faire pour s'assurer de gagner ?

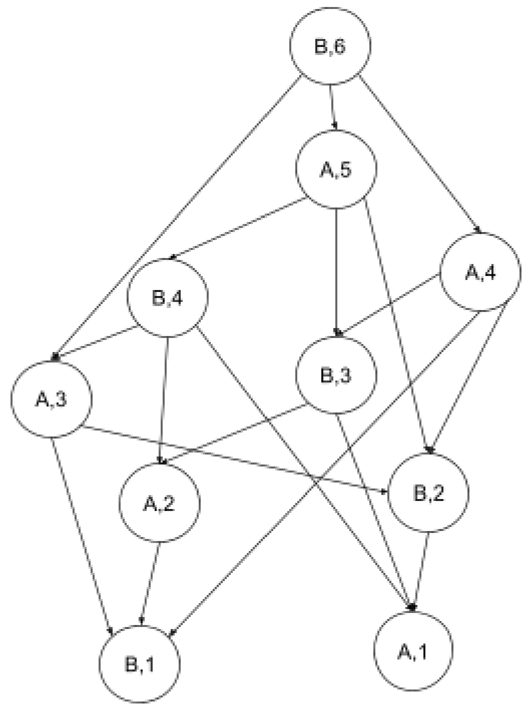


Figure 6.1