



**Université  
Paris Cité**

---

**Rapport de Final  
Projet d'intelligence artificielle  
Othello/Reversi  
2024-2025**

---

**Groupe TD : n° 01**

**Mohamed El Amine MAZOUZ**

**Mohamed OUBERKA**

## Description du Jeu et de ses Règles

Othello, également connu sous le nom de Reversi, est un jeu de stratégie pour deux joueurs joué sur un plateau de 8x8 cases. Les joueurs utilisent des pions bicolores (Noir et Blanc). L'objectif est de terminer la partie avec le plus grand nombre de pions de sa couleur sur le plateau.

## Règles principales

1. Chaque joueur joue à tour de rôle, en plaçant un pion de sa couleur sur une case vide.
2. Un coup est valide s'il encadre au moins un ou plusieurs pions adverses entre le pion joué et un autre pion de la même couleur.
3. Les pions adverses encadrés sont retournés pour devenir de la couleur du joueur actif.
4. Si un joueur ne peut pas jouer de coup valide, il passe son tour.
5. La partie se termine lorsque le plateau est plein ou qu'aucun joueur ne peut jouer. Le joueur avec le plus de pions de sa couleur gagne.

## Descriptif des IA implémentées

Trois niveaux d'IA ont été développés pour ce projet : EasyAI, MediumAI, et HardAI. Voici une description détaillée de chaque IA et des choix effectués :

### EasyIA :

- Profondeur de recherche : 1 (Minimax simple).
- Heuristique utilisée : Évaluation basique basée sur la différence de pions, le contrôle des coins et des bords.
- Choix justifié : Cette IA est conçue pour être rapide et offrir un défi minimal, idéale pour les débutants.

### MediumAI

- Profondeur de recherche : 4.
- Techniques supplémentaires : Limite de temps de réflexion (10 secondes). Ordre des coups basé sur la priorité des coins et des bords.
- Heuristique utilisée : Évaluation simple avec prise en compte de la mobilité.
- Choix justifié : Cette IA offre un équilibre entre performance et complexité, adaptée aux joueurs intermédiaires.

### HardAI

- Profondeur de recherche : Approfondissement itératif jusqu'à 6.
- Techniques supplémentaires : Limite de temps de réflexion (10 secondes). Table de transposition. Ordre des coups avancé.

- Heuristique utilisée : Prise en compte de la stabilité, de la mobilité et de la parité.
- Choix justifié : Conçue pour offrir un défi maximal, simulant un joueur expert.

## Tournoi entre IA : analyse des résultats

Afin de mesurer l'efficacité des différentes versions de notre intelligence artificielle pour le jeu Othello, nous avons mené un tournoi automatisé en confrontant les trois niveaux disponibles : facile, moyen et difficile.

Chaque duel a donné lieu à 50 parties, en alternant l'ordre de jeu pour garantir l'équité.

### Facile vs Moyen

- Victoires : 0 – 50 (en faveur de l'IA moyenne)
- Écart total de disques : –2325 pour l'IA facile
- Moyenne de disques : 2,5 / 64 (facile) contre 49,0 / 64 (moyenne)

### Facile vs Difficile

- Victoires : 0 – 50 (en faveur de l'IA difficile)
- Écart total de disques : –2250 pour l'IA facile
- Moyenne de disques : 9,5 / 64 (facile) contre 54,5 / 64 (difficile)

### Moyen vs Difficile

- Victoires : 0 – 50 (en faveur de l'IA difficile)
- Écart total de disques : –660 pour l'IA moyenne
- Moyenne de disques : 25,3 / 64 (moyenne) contre 38,5 / 64 (difficile)

## Interprétation

Les résultats montrent une progression nette de performance avec la profondeur de recherche.

L'IA facile ne parvient à remporter aucune partie, quels que soient ses adversaires. L'IA moyenne, bien que largement battue par la version difficile, affiche des résultats plus compétitifs, notamment en termes de disques moyens par partie.

L'IA difficile, grâce à une profondeur de recherche plus importante, surpasse systématiquement les autres niveaux. Ces résultats valident le lien direct entre la profondeur de l'algorithme Minimax, la qualité des décisions, et le résultat final dans un jeu comme Othello, à information complète.

## Structure du Code

- ``othello_game.py`` : logique du jeu (règles, placements, vérifications).
- ``ai_strategies.py`` : implémentations des différentes IA.
- ``tournament.py`` : matches IA vs IA.
- ``othello_launcher.py`` : lancement d'une partie utilisateur vs IA.
- ``readme.md`` et ``LICENSE`` : documentation et licence du projet (GitHub).

## Bilan du Projet

Ce projet nous a permis d'explorer en profondeur les concepts liés à la prise de décision autonome, aux fonctions d'évaluation, et aux algorithmes de recherche dans les jeux à deux joueurs.

Le choix d'Othello s'est révélé pertinent : bien que les règles soient simples, le jeu présente une richesse stratégique importante.

Les IA développées illustrent clairement l'impact des différentes techniques d'optimisation, et les résultats du tournoi ont confirmé la cohérence de nos choix d'implémentation.

## Références

### Rapport

- ChatGPT 4-o : Nous avons utilisé ChatGPT tout au long de la rédaction du rapport pour nous assister dans la formulation, la relecture, la correction orthographique et la clarification des idées. L'outil nous a permis de mieux structurer les différentes parties du document, notamment lors des descriptions techniques.

### Développement

- ChatGPT 4-o : ChatGPT nous a également aidés durant le développement du projet, en particulier pour :
  - comprendre certains comportements inattendus de l'IA,
  - corriger des bugs complexes,
  - structurer l'algorithme Minimax et intégrer l'élagage alpha-bêta de manière efficace.
- GitHub Copilot : Copilot a été utilisé comme assistant de codage pour suggérer des complétions, générer des structures de fonctions et accélérer la mise en place de la logique du jeu et de l'interface.