



**1**

Règles

**2**

Structures de  
données

**3**

Algorithmes

**4**

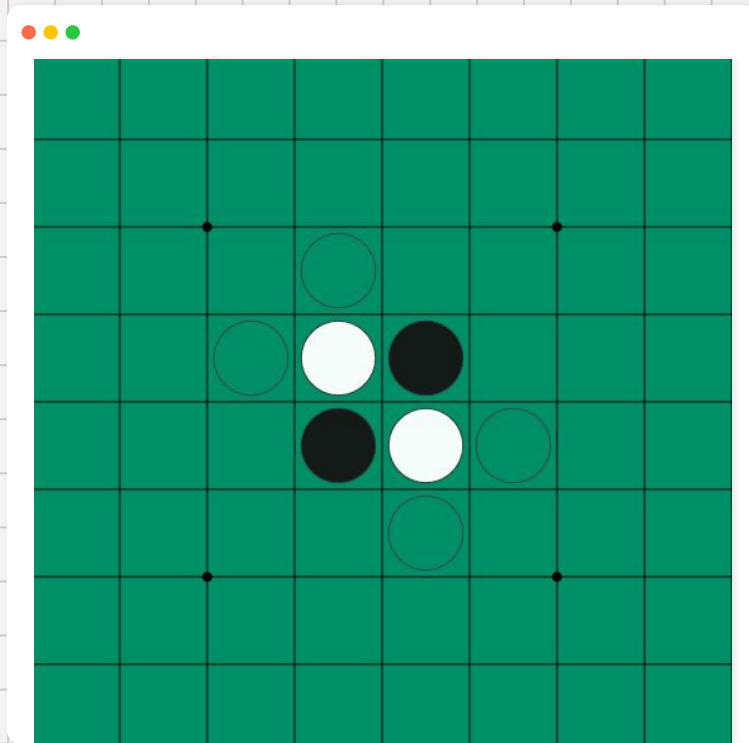
Besoins visés

**5**

Agenda  
prévisionnel

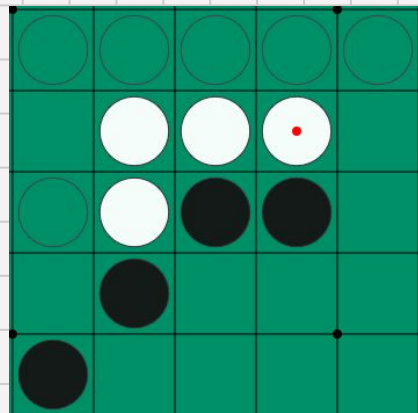
# Explication des règles

- Othello se joue sur un plateau de taille 8x8
- 2 jetons blancs et noirs sont disposés au centre au début de la partie. Les noirs commencent.
- Les joueurs jouent à tour de rôle un coup, chaque coup doit capturer au moins un jeton pour être valide.
- Pour capturer un pion adverse, il faut l'encercler de 2 jetons.
- Pour gagner il suffit d'avoir plus de jetons de sa couleur une fois le tableau rempli ou que les 2 joueurs sont à cours de coups.

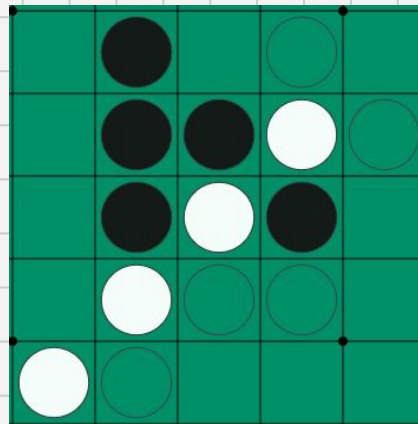


# Précision des règles

1.



2.



## Jouer un coup (Noirs):

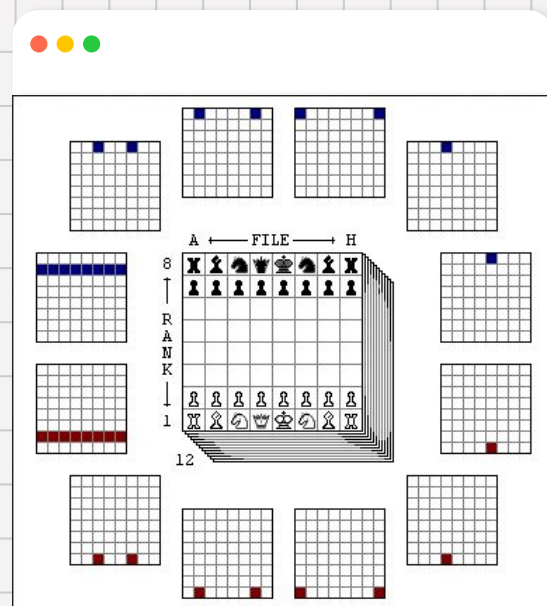
- Enfermer les pièces adverses entre deux pièces alliés
- Nous pouvons enfermer les pièces sur les lignes, collones, et diagonales.

# Structures de données

## Bitboards

Utilisation de 5 “mots binaires”:

- Pour le board rempli entièrement (masque)
- Pour les pions (un pour les pions noirs et un pour les pions blancs)
- Deux masques pour éviter les débordements à droite et à gauche lors des shift (un par côté)



# Structures de données

Bitboards  
(masquage)

-a-b-c-d-e-f-g-h-								
								1
								2
								3
	B		W					4
	B			W				5
								6
								7
								8

Position initiale

-a-b-c-d-e-f-g-h-								
								1
								2
								3
	B			W			B	4
				W				5
								6
								7
								8

Overflow à cause d'un shift

-a-b-c-d-e-f-g-h-								
								1
								2
								3
	B			W				4
				W				5
								6
								7
								8

Masquage

# Structures de données

Fichiers de sauvegarde

- Système de sauvegarde et d'historique en ASCII
- Fichiers de configuration au format ini

```
[Default]
size = 8
ai = false
ai-color = W
ai-mode = minimax
ai-depth = 3
ai-heuristic = default
ai-time = 5
ai-level = medium
```

# Algorithmes utilisés

Algorithme tiré de :  
**Bitboard methods for games de  
Cameron Browne**



## Calcul des coups légaux - Algorithme Line Cap Moves

Génère un ensemble de coups possibles dans chaque direction.

Algorithme:

- 1- Initialiser un ensemble de moves vide
- 2- Itérer sur toutes les directions: N, NE, E, SE, S, SW, W, NW
- 3- Générer un ensemble de coups 'candidats'
- 4- Tant que 'candidats' n'est pas un ensemble vide , faire:
  - 5- Actualiser 'moves' si lorsque l'on décale de 1, la case est vide
  - 6- Réécrire 'candidat' en décalant de 1 dans la direction

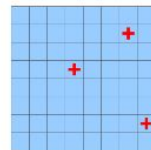
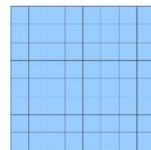
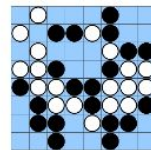
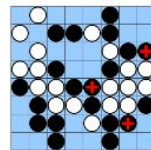
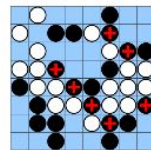


Illustration de Line Cap Moves sur la direction Est  
pour le joueur aux pions blancs



# Algorithmes utilisés

Algorithme basé sur Line Cap Moves, tiré de :  
**Bitboard methods for games de Cameron Browne**

## Capture des pions adverses

A partir du dernier pion placé, change la couleur des pions capturés.

Algorithme:

- 1- Récupérer le dernier pion placé par le joueur courant, noté 'pion'
- 2- Itérer sur toutes les directions: N, NE, E, SE, S, SW, W, NW
- 3- Décaler 'pion' de 1 case sur la direction courante (sur un pion Ennemi)
- 4- Tant que 'pion' n'est pas un pion Ami, faire:
  - 5- Modifier le 'pion' en pion Ami
  - 6- Décaler 'pion' de 1 sur la direction courante

# Algorithmes utilisés

Algorithmes utilisés pour  
l'implémentation du joueur IA

## Algorithme MiniMax

Algorithme de parcours de l'arbre des coups possibles.  
Parcours l'arbre jusqu'à une profondeur donnée.  
Remonte le meilleur coup évalué par l'heuristique choisie.

## Algorithme Alpha-Beta pruning

Algorithme de même type que MiniMax.  
Effectue un élagage des branches inutiles à l'évaluation des coups.

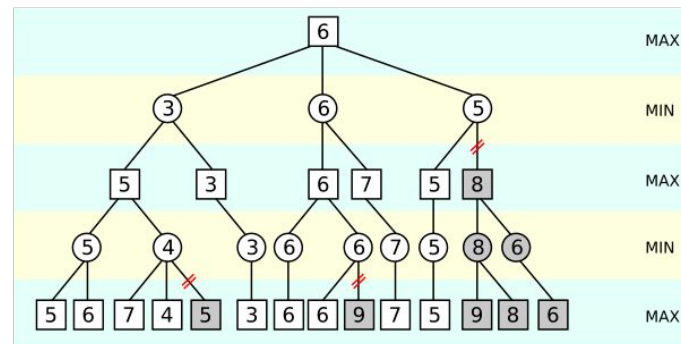


Illustration d'un élagage Alpha-Beta, tiré de  
Wikipedia:  
[https://upload.wikimedia.org/wikipedia/commons/thumb/9/91/AB\\_pruning.svg/2880px-AB\\_pruning.svg.png](https://upload.wikimedia.org/wikipedia/commons/thumb/9/91/AB_pruning.svg/2880px-AB_pruning.svg.png)

# Besoins visés

## Généralités



- Python 3.7
- PEP8
- Fonctionne avec GNU/Linux
- Documentation
- Tests / Coverage
- Calculs des performances

F.1-F.8

## Bibliothèques et outils



- Build System pyproject et setuptools
- Framework de tests
- Gestion des options
- PyGObject

F.9-F.12

# Besoins visés

## Options en ligne de commande



- Nom de l'exécutable
- Usage
- Aide
- Version
- Mode debug
- Taille du plateau
- Mode blitz
- Mode Contest
- Mode IA

F.14-F17, F.19-F.24

## Interface Utilisateur



- Interface ligne de commande
- Interface graphique
- Notation des coup
- Historique
- Messages d'erreurs
- Quitter et Sauvegarder
- Sauvegarde de l'historique
- Recommencer / abandon / Rejouer une partie

F.25-F.33, F.35-F.37

# Besoins visés

## Formats d'entrée/sortie



- Format de fichier simplifié
- Format de l'historique
- Fichier de configuration

F.38-F.40

## Structures internes



- Plateau de jeu
- Module bitboard
- Etat du jeu

F.41-F.43

# Besoins visés

## Interface Graphique



- Fonctions de base
- Affichage des coups jouées

F.44, F.49

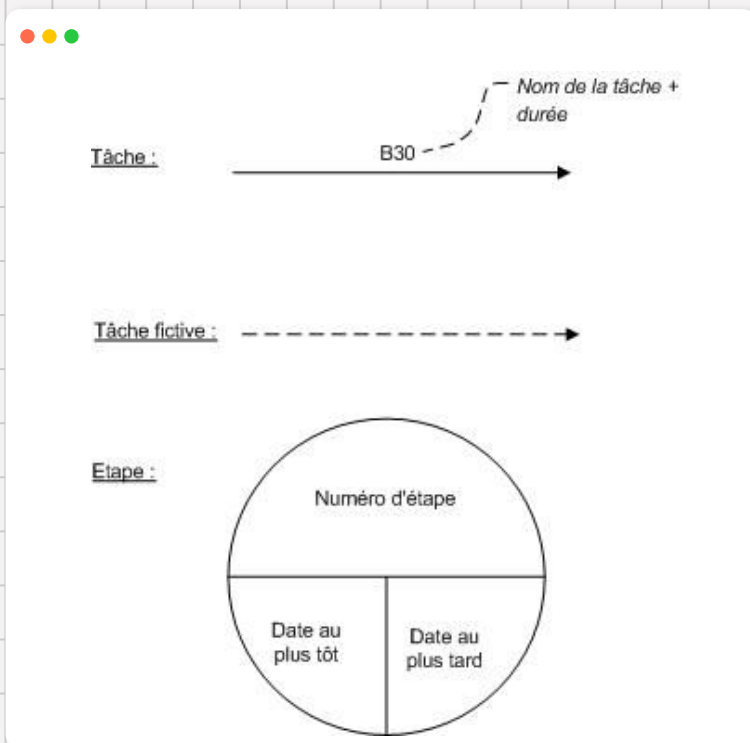
## Joueur artificiel



- Heuristiques
- Minimax
- Alpha-beta pruning
- Exploration peu profonde préliminaire
- Profondeur de recherche
- Choix des heuristiques
- Calculs des libertés
- Temps de réflexions bornée

F.51-F.54, F.56-F.58, F.60

# Répartition des tâches



## Gestion de projet avec PERT:

- Décrire, représenter, analyser, et suivre de manière logique les tâches
- Avoir une date précise pour chaque étape de la conception.
- Suivi d'un chemin critique pour les tâches éligibles.

# Semaine 1

Date

Besoins

Resources 10h/Semaine

Lundi  
5/2

- Méta actions: F.2, F.6, F.10
- Depot configuré: F.4

- Lucas Marques, 3 heures
- Gabriel Tardiou, 2 / 10 heures
- Matis Duval, 2 / 5 heures
- Lucas Zammit, 3 heures

Mercredi  
7/2

- Depot configuré: F1, F.9

- Remy Heuret, 4 heures
- Lucas Marques, 1 heure
- Matis Duval, 3 / 5 heures
- Gabriel Tardiou, 3 / 10 heures

Vendredi  
9/2

- Squelette du projet: F.11, F.12, F.16, F.17

- Gabriel Tardiou, 5 / 10 heures
- Remy Heuret, 6 heures
- Lucas Zammit, 5 heures
- Matis Duval, 5 heures
- Lucas Zammit, 5 heures

Milestone 1: Acomplie



# Semaine 2

Date

Besoins

Resources 10h/Semaine

Lundi

- Module Bitboard, F.42
- Logger configurable, F.19

- Lucas Marques
- Lucas Zammit

Mercredi

- Module configurable, F.22, F.21, F.23, F.24

- Gabriel Tardieu
- Matis Duval
- Rémy Heuret
- Lucas Zammit

Vendredi

RATTRAPAGE DES BESOINS

RATTRAPAGE DES BESOINS

Milestone 2: EN COURS..

# Semaine 3

Date

Besoins

Resources 10h/Semaine

Lundi

- IA Configurable, F.57, F.54, F.56,F.60

- Rémy Heuret
- Lucas Zammit

Mercredi

- Partie Configurable, F.26, F.15, F.40

- Gabriel Tardieu
- Matis Duval
- Rémy Heuret

Vendredi

- Partie Jouables, F.41, F.43

- Lucas Marques
- Lucas Zammit

Milestone 2: Acomplie

# Semaine 4

Date

Besoins

Resources 10h/Semaine

Lundi

- Affichage Partie, F.49
- Sauvegarde d'une partie, F.39, F.38

- Matis Duval
- Gabriel Tardiou
- Lucas Marques

Mercredi

- Chargement d'une partie, F.39,F.38

- Lucas Zammit
- Matis Duval

Vendredi

RATTRAPAGE DES BESOINS

RATTRAPAGE DES BESOINS

Milestone 3: EN COURS..

# Semaine 5

Date

Besoins

Resources 10h/Semaine

Lundi

- Module Bitboard, F.42
- Logger configurable, F.19

- Lucas Marques
- Lucas Zammit

Mercredi

- Module configurable, F.22, F.21, F.23, F.24

- Gabriel Tardieu
- Matis Duval
- Rémy Heuret
- Lucas Zammit

Vendredi

RATTRAPAGE DES BESOINS

RATTRAPAGE DES BESOINS

Milestone 3: EN COURS..

# Semaine 6

Date

Besoins

Resources 10h/Semaine

Lundi

- Interface CLI, F.28, F.36, F.32, F.30, F.35

- Matis Duval
- Gabriel Tardieu
- Lucas Marques
- Rémy Heuret

Mercredi

- Interface CLI, F.29, F.33, F.37, F.27, F.31

- Gabriel Tardieu
- Matis Duval
- Rémy Heuret
- Lucas Zammit

Vendredi

RATTRAPAGE DES BESOINS

RATTRAPAGE DES BESOINS

Milestone 3: Acomplie

# Semaine 7

Date

Besoins

Resources 10h/Semaine

Lundi

- Heuristiques fonctionnelles, F.51, F.58
- AlphaBeta pruning, F.53

- Rémy Heuret
- Lucas Zammit

Mercredi

- Minimax, F.52
- IA Fonctionnelle, F.60

- Lucas Marques
- Matis Duval

Vendredi

RATTRAPAGE DES BESOINS

RATTRAPAGE DES BESOINS

Milestone 4: En cours...

# Semaine 8

Date

Besoins

Resources 10h/Semaine

Lundi

- Testing + Benchmarking + Documentation, F.8, F.7, F.3, F.5

- Gabriel Tardiou
- Lucas Zammit
- Rémy Heuret
- Lucas Marques

Mercredi

RATTRAPAGE DES BESOINS

RATTRAPAGE DES BESOINS

Vendredi

Milestone 4: Acomplie