

# Projet de l'UE Programmation C

**Préliminaire :** Le but de ce projet est de construire un jeu de reversi (ou Othello). Le projet vient avec différents niveaux qui permettent un nombre croissant d'options. **Il est demandé le jeu de base plus au moins 3 options.** C'est la demande minimale. Le projet est prévu pour des groupes de 2 étudiant(e)s, exceptionnellement 3. Il se soldera par un rapport, une présentation orale avec des transparents et une démonstration de votre logiciel (voir à la fin de ce sujet). Au préalable, il sera demandé les codes, un court rapport expliquant les choix d'implémentation et les différentes parties implémentées, ainsi qu'une présentation générale de votre logiciel et jeu. Le calendrier précis vous sera donné dans les prochaines semaines.

## Jeu de base :

Le plateau de jeu est une grille de  $N \times N$  cases (Note : plus  $N$  est grand, plus la partie durera longtemps). Une case peut être vide ou bien contenir un pion vert ou un pion rouge. Comme on le verra par la suite, certaines cases peuvent être piégées.

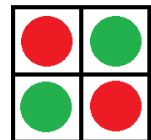
Le jeu de base se joue à 2 joueurs (mais on peut imaginer jouer à plus, cf. niveau 5). Un nombre  $M$  de cases piégées sont prévues ( $M$  est aussi un paramètre). Les cases piégées sont déterminées de manière aléatoire et non connues par les joueurs. A chaque joueur est attribuée une couleur (Vert ou Rouge), les joueurs jouent chacun à leur tour.

## But du jeu :

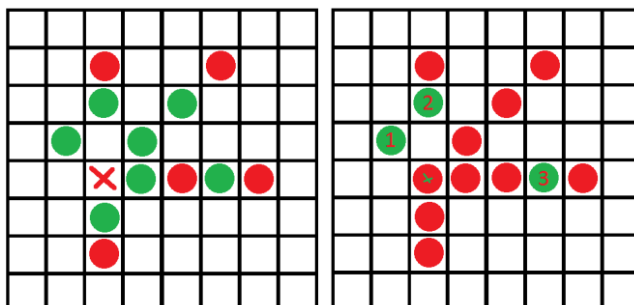
Le jeu s'arrête quand plus personne ne peut jouer. L'objectif de chaque joueur est d'avoir un maximum de pions de sa couleur sur le plateau. Au cours de la partie, les pions peuvent changer de joueur (cf. après). À la fin du jeu, le joueur qui a le plus de pions de sa couleur sur la grille a gagné.

## Déroulement du jeu :

- Au début d'une partie, une configuration de quatre pions de départ (voir image) est placée soit au centre, soit, pour pimenter un peu le jeu, aléatoirement quelque part sur le plateau (évités de les coller au bord ou dans un coin). Attention également à ne pas placer une bombe trop près des pions de départ.
- A son tour, un joueur doit poser un pion (de sa couleur) sur le plateau. Pour cela, il doit rester une case vide sur le plateau, et la pose du pion (sur une case vide) doit entraîner le changement de couleur d'au moins un pion adverse. Pour cela, il faut que la pose de son pion « enferme » une ligne contiguë de pions adverse : tous les pions adverses qui se trouvent entre le nouveau pion posé et un autre pion de même couleur (horizontalement, verticalement ou en diagonale) prennent alors la couleur du joueur qui vient de poser son pion. Il ne faut pas de case vide entre le pion posé et le prochain pion de même couleur.



Exemple :

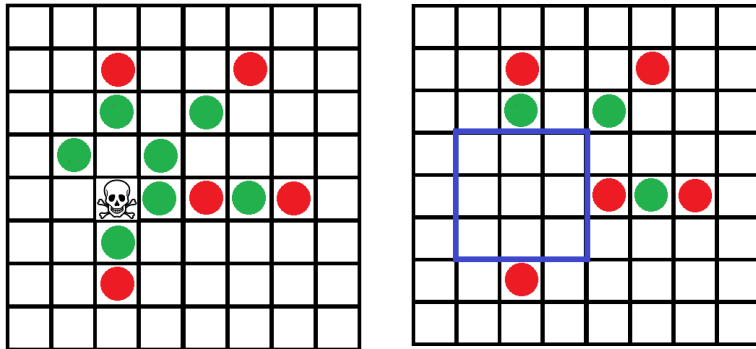


Dans ce cas, si le joueur rouge place un pion sur la case marquée d'une croix, tous les pions verts sauf trois changent de couleur. Le pion (1) n'est pas « enfermé » entre deux pions rouges. Le pion (2) non plus, (il y a une case vide entre le pion posé et le rouge au-dessus de lui, or il faut que la suite de pions soit ininterrompue. Le pion vert (3) n'est pas

changé car il y a un pion rouge entre lui et le pion posé. Seul le premier pion vert sur la même ligne a donc changé de couleur.

- Si le joueur pose un pion sur une case piégée, une bombe explose. Dans la version de base du jeu, les 8 pions autour de la case sont détruits. Si la case piégée est en (x,y) alors les pions des cases (x±1, y±1) sont enlevées du plateau. La case piégée ainsi que les 8 cases autour de la bombe sont à présent disponibles (i.e. vides). La bombe une fois explosée est retirée du jeu et n'explose pas une seconde fois.

Exemple : imaginons cette fois-ci que la case marquée d'une tête de mort cache une bombe (là on a marqué la place de la bombe pour l'exemple, mais le joueur bien sûr ne sait pas que cette case est piégée quand il joue). Si un joueur pose un pion sur cette case, les pions des huit cases autour sont enlevés du plateau (les pions dans le carré bleu de la deuxième image ont disparu). Attention à ne pas placer de bombe trop près des cases de départ où vous risquez de bloquer la partie trop tôt.



#### Améliorations et options de jeu supplémentaires :

##### Bombes artisanales :

- On peut imaginer différents types de bombes en plus de celle mentionnée dans la version de base:
  1. Une bombe qui explose et qui détruit la case de la bombe. La case devient injouable (attention à bien gérer les conditions de fin de partie) ;
  2. La bombe fait disparaître les pions des cases autour de la bombe, mais le pion posé reste sur place (il ne disparaît pas dans l'explosion) ;
  3. Une case piégée qui change la couleur du pion qui vient d'être posé ;
  4. Une case piégée qui avec son laser ultra puissant élimine tous les pions dans une (au hasard) des 8 directions possible autour de la case piégée.
  5. ... à votre imagination...

##### Un traître parmi nous :

- L'idée de base est qu'un pion n'a pas forcément éternellement la même couleur ; il peut changer de camp tout seul, le traître. On attribue à chaque pion un âge qui correspond au nombre de coups joués depuis qu'il a été posé ou depuis la dernière fois qu'il a changé de couleur (donc quand un pion change de couleur, par trahison ou selon les règles de base, son compteur d'âge redémarre à zéro). Plus un pion est vieux, plus sa probabilité de changer de camp est élevée.
- Quand un pion change de camp (le traître), on applique les règles de changements de couleurs comme s'il s'agissait d'un pion qui vient d'être posé.
- On gère la phase de trahison pour les pions d'un joueur juste avant son tour de jeu (ou juste après. A vous de voir).
- La gestion de la trahison peut s'envisager sous deux méthodes :
  1. Vous attribuez un % de chance de devenir un traître à chaque pion selon son âge (nombre de coups depuis qu'il a été posé ou qu'il a changé la dernière fois de couleur). Zéro, un ou plusieurs pions peuvent donc changer de camp en même temps. Attention

à bien choisir vos % car sinon soit jamais rien ne se passera ou alors, on aura tellement de trahisons que le jeu sera trop hasardeux et donc injouable.

2. Une autre possibilité, peut-être plus jouable, est de considérer qu'à chaque tour on aura (forcément) un et un seul traître. Avec toujours la règle que, plus le pion est vieux, plus il a de chances d'être le traître (la somme des % de tous ses pions fait 1). Evidemment, au départ, les premiers pions que l'on pose ont de fortes chances de changer de camp (en particulier le premier pion) et si l'on est malin, on inclut cela dans sa stratégie de jeu.

### **Cartes événements :**

- Chaque joueur se voit attribuer, en début de partie, trois à cinq cartes événements (selon la taille du plateau de jeu). Chaque carte correspond à un événement affectant la partie.
- A son tour de jeu, un joueur peut choisir de poser un pion ou de jouer une de ses cartes événements (jusqu'à épuisement de son stock de cartes).

Les cartes peuvent avoir différents effets (ci-dessous quelques exemples. Vous pouvez bien sûr en imaginer d'autres, pleins d'autres).

- Changer de camp (le joueur qui avait les pions rouges a désormais les pions verts et réciproquement. Suite à cette action, le joueur devra donner la carte de son choix (il doit donc en posséder encore au moins une) à son adversaire.
- Changer un pion de son adversaire en un pion de sa propre couleur (avec ou sans appliquer les règles de changement de couleur),
- Retirer un pion adverse du plateau (la case est vidée et un nouveau pion pourra y être joué)
- Jouer deux fois de suite,
- Créer un trou dans le plateau (aucun pion ne pourra plus y être déposé. Un joueur ne peut créer un trou que sur une case ne contenant pas de pion),
- Être immunisé contre les effets de la prochaine bombe sur laquelle on tombera (il faut jouer la carte pour qu'elle prenne effet). Cela ne concerne pas les effets des bombes que déclencherait votre adversaire.
- Possibilité de poser un pion sur n'importe quelle case vide.
- Etc, etc, etc.

## **Jeux et options**

### **Projet de base. Version minimale demandée :**

- Le jeu de base décrit ci-dessus avec les bombes cachées sur le plateau. Une gestion de l'affichage de base se fera en mode texte. On suppose ici que le plateau de jeu a une taille fixe et que le nombre de bombes à cacher est demandé en début de partie.

### **Option 1 : « bombes artisanales », et plateau de différentes tailles :**

- Une gestion de différents types de bombes. On doit proposer aux joueurs trois tailles de plateaux selon que les joueurs veulent une partie courte, moyenne ou longue. Le nombre de bombes doit être adapté à chaque taille de plateaux.

### **Option 2 : « un traître parmi nous » :**

- Gestion de l'option « un traître parmi nous ».

**Option 3 : Projet avec les cartes événements :**

- On demandera ici de gérer le jeu avec les cartes événements. Libre à vous de considérer également la gestion de « un traître parmi nous », mais on continue à offrir plusieurs types de parties (courte, moyenne, longue), plusieurs types de bombes.

**Option 4 : Plus de deux joueurs**

- On proposera un jeu à plus de 2 joueurs (à vous d'adapter les règles, en particulier la position des pions de départ).

**Option 5 : Chacun joue sur son jeu**

- Un bon petit « Fork » et un fichier commun accessible par les deux programmes qui permette à chaque joueur de jouer sur sa propre interface. On pourra ainsi envisager d'afficher des informations différentes pour chaque joueur (imaginons que certaines bombes soient visibles pour l'un et pas pour l'autre... des bombes que l'on aurait au hasard posé soi-même à l'aide d'une carte événement).

**Option 6 : Aide de l'ordinateur**

- On proposera une aide de l'ordinateur qui vous proposera un meilleur coup (sans tenir compte des cartes) et la possibilité de revenir un coup en arrière.

**Option 7 : Jouer contre l'ordinateur**

- Vous jouez contre l'ordinateur ou l'ordinateur joue contre lui-même. Attention à voir si cette possibilité est compatible avec d'autres options comme les cartes événements, etc. Vous pouvez toujours donner la possibilité à deux joueurs de jouer l'un contre l'autre avec toutes les options ou de jouer contre l'ordinateur en désactivant certaines options (ou pourquoi pas, en les autorisant mais avec un tirage aléatoire).

**Option Scramble :** vous pouvez bien sûr mélanger les options, proposer l'option 3 avec 4 joueurs, etc. Veuillez simplement à bien nous expliquer, dans votre rapport et présentation, à quel jeu on joue. Vous gagnez si on s'amuse à jouer à votre jeu et qu'on comprenne comment vous avez programmé le jeu.

**Options :** les éléments suivants sont optionnels : Gestion graphique (affichage du plateau, des cartes...), jouer à distance entre deux ordinateurs.

## **LIVRABLES ATTENDUS :**

**Rapport, Code et Soutenance :**

- Nous demandons un rapport d'une quinzaine de pages maximum qui détaille le jeu que vous proposez (quelles règles, quel niveau, etc.), vos choix, la structure générale de votre code (les grands blocs, les bibliothèques que vous avez utilisé, etc.). Le rapport doit nous servir de guide pour lire votre code et comprendre le jeu que vous proposez. Terminez le rapport avec quelques commentaires sur les difficultés rencontrées, etc.

- **Le code doit être commenté.** Au minimum, on doit avoir un commentaire décrivant ce que fait chaque fonction et procédure. **Attention, la qualité du code (lisibilité, choix des noms de variables et fonctions, indentation, complexité des fonctions...) sera largement prise en compte dans la notation. Un programme de qualité, même sommaire, peut obtenir plus de points qu'un programme illisible comprenant toutes les options.**

- La soutenance devra comporter quelques slides de présentation (jeu, structure du code) et une démonstration.