

# Algorithmique/Python

Master “Technologie et Handicap” : Intensifs d’Automne

Examen du 6 octobre 2012, 17h

Durée 2h, documents (poly et notes de cours) autorisés

## 1 Fonctions et tableaux

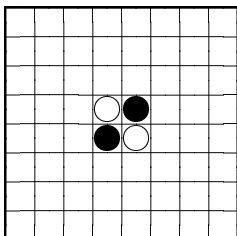
- On souhaite écrire une fonction qui renvoie un nombre donné arrondi à un nombre de décimales déterminé.
  - Quel(s) est(/sont) l’(/les) argument(s) de cette fonction ?
  - Que renvoie cette fonction ?
  - Écrire la fonction. On pourra utiliser la fonction *partieEntiere* :  
`partieEntiere(var v réel var valeur) : entier`
  - Implémenter cette fonction en Python  
Rappel : la fonction *partieEntiere* en python est : `def int (v)`
- On souhaite maintenant écrire une fonction permettant d’arrondir tous les éléments d’un tableau (*en utilisant la fonction précédente, bien sûr*).
  - Quel(s) est(/sont) l’(/les) argument(s) de cette fonction ? (en particulier comment cet(/ces) argument(s) est(/sont)-il(s) passé(s)).
  - Que renvoie cette fonction ?
  - Écrire la fonction.
  - Implémenter cette fonction en Python

## 2 Réversi

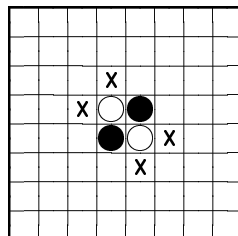
Le jeu de *Réversi* est inspiré du jeu chinois de Go (le plus ancien jeu de stratégie connu). Deux joueurs s’affrontent avec des pierres blanches et noires sur un plateau de 8 x 8 cases.

Le principe du jeu est simple. Un joueur doit poser sa pierre à l’extrémité d’une ligne de pierres adverses et dont l’autre extrémité comporte déjà un de ses propres pions. Les pions ainsi encadrés sont alors pris à l’adversaire (et remplacés par des pierres de la couleur de celui qui vient de prendre). S’il ne peut rien prendre, le joueur doit passer son tour. Dans le jeu de *Réversi* les lignes peuvent être horizontales, verticales ou diagonales. Pour simplifier **on ne considérera ici que les lignes horizontales et verticales**.

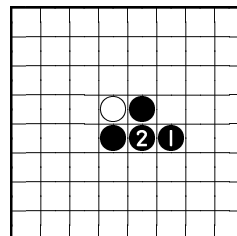
Voici la position de départ et les mouvements possibles.



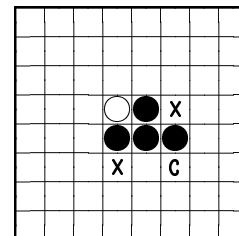
Position de départ



Mouvements possibles pour les noirs



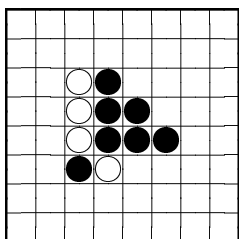
Les noirs jouent 1, la pièce 2 est prise



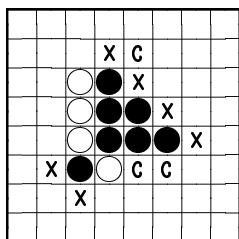
Mouvements possibles pour les blancs

**NB** : On a marqué d’un C les mouvements possibles dans le jeu *Réversi*, mais qu’on ne prendra pas en compte pour simplifier, comme indiqué plus haut (lignes en diagonales).

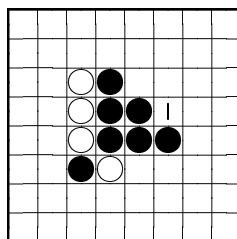
Dans cet exemple un peu plus compliqué c'est au blancs de jouer :



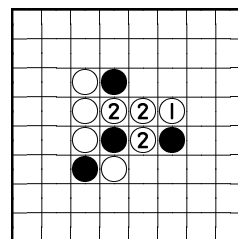
Situation initiale, les noirs viennent de jouer



Mouvements possibles pour les blancs



Les blancs jouent 1



Les pièces 2 sont prises

On représente le plateau de jeu par un tableau de caractères à deux dimensions, pour chaque case on utilisera le caractère *espace* si la case est vide, le caractère 'b' si elle contient un pion blanc, et le caractère 'n' si elle contient un pion noir.

```
var reversi: car[8,8]
```

Questions :

1. Écrire une fonction qui initialise le plateau de jeu, selon la figure « Position de départ ».
2. Écrire une fonction qui renvoie vrai si une case du plateau, dont les coordonnées sont données en arguments, est libre, et faux sinon.
3. Écrire une fonction qui compte le nombre de cases occupées par une des couleurs donnée en argument.
4. Écrire une fonction qui compte dit qui a le plus de pions sur le plateau de jeu (et donc qui a gagné si le jeu est terminé).
5. Écrire une fonction qui vérifie si un joueur peut jouer sur une case dont les coordonnées sont données en arguments.

### Question bonus

6. Écrire une fonction qui « joue » un coup, c'est à dire qui place le pion dans le plateau de jeu et change de couleur les pierres prises.

Attention : tous les arguments n'ont pas été donnés dans les énoncés.

### Barème indicatif

Partie 1 (Fonctions et tableaux) : 4/4, Partie 2 (*Réversi*) : 2/2/2/2/4/+2

Attention dans la seconde partie (*Réversi*) 25 % des points seront attribués si la déclaration de la fonction est correcte (avec les bons arguments).