TP2 : Le morpion (Tic Tac Toe)

Objectif du TP

Réaliser un jeu de morpion (tic tac toe) en scheme, à deux joueurs humains, sans interface graphique.

1) regarder la décomposition fonctionnelle proposée dans le fichier morpion-vide.rkt
2) implémenter chacune des fonctions

Pour les plus courageux (For Honor and Glory) : interface utilisateur graphique, IA, ...

A déposer dans la boîte de de dépôt.

Elements de cours :

La structure de base pour le plateau de jeu est vector. Ce sont des tableaux à une dimension. Ce sont des objets *mutables* (par défaut : il est possible de les rendre *immutables*) c'est à dire qu'il est possible de les modifier. Les opérations usuelles sur les vectors sont:

- (vector? v) : renvoie #t si v est un vector
- (make-vector size [v]) : crée un vector de taille size, éventuellement initialisé avec la valeur v
- (vector v ...) : crée un vector assez grand pour y placer tous les paramètres v
- (vector-ref vec pos) : renvoie la valeur à la position pos (numérotée à partir de 0) dans le vector vec
- (vector-set! vec pos v) : change (oui ! c'est une fonction à effet de bord : l'objet est changé ; par convention le nom de ce type de fonctions est suffixé par !) par v la valeur du vector vec à la position pos (numérotée à partir de 0).

Afin de conserver un style fonctionnel, vous pouvez encapsuler vector-set! :

```
(define vector-modif vec pos val)
  (vector-set! vec pos val)
  vec
)
```

Trivia :

- Tic-Tac-Toe : une façon amusante de représenter l'arbre des coups optimaux du morpion
- Wargames : le morpion joue un rôle décisif pour le dénouement de ce film ...