

RAPPORT PUISSANCE 4

I. Les Fonctionnalités

J'ai commencé par coder la partie a 2 joueurs [Humain contre Humain] pour avoir quelques fonctions de base tel que la vérification de 4 pions aligné [Défini par une constante pour la modularité]. Dans cette partie 2 joueurs les 'X' commencerons toujours [par définitions du puissance 4 les pions de couleur rouge commence ici j'ai choisi arbitrairement les 'X'].

Dans ce mode les grilles peuvent être sauvegardées pour y rejouer plus tard.

Après avoir lu (beaucoup) de document en rapport avec les Intelligence Artificielle j'ai décidé de coder dans un premier temps l'algorithme Min-Max réussi avec quelque difficulté. J'ai donc commencé mes tests mais j'ai remarqué que les temps d'attente étaient un peu trop importants [ex : 15 secondes d'attente avec une profondeur de 9].

J'ai donc continué mes recherches et j'ai trouvé un moyen (simple?) d'améliorer mon algorithme Min-Max avec l'élagage Alpha – Bêta ; Le gain de temps a été énorme sur le temps de réponse de l'IA [passant de 15 s avec une profondeur de 9 à un temps de réaction de 1, 5 seconde environ].

Cette Intelligence Artificielle m'a permis ainsi de créer 2 autres types de jeux :

- Ordinateur contre humain
- Ordinateur contre Ordinateur [Un mode démo au final]

Ayant quelque idée d'améliorations en tête, je me suis amusé à coder quelques fonctionnalités en plus tel que la gestion de statistiques [21 statistiques répertoriées dans un fichier texte], l'affichage des règles [Les règles sont simplement contenues dans un fichier texte qui est récupéré et affiché].

Les statistiques peuvent être réinitialisées à partir du menu OPTIONS.

Le menu OPTIONS n'est composé que d'une seule car malheureusement je n'ai pas eu le temps de coder la seconde option que je voulais faire à savoir supprimer une partie enregistrée [la méthode que j'aurais voulu utiliser était de recopier le dossier dans un dossier temporaire jusqu'à la ligne que l'on veut supprimer puis continuer et enfin recopier le fichier texte temporaire dans le fichier de sauvegarde du plateau de jeu].

L'autre amélioration ajoutée est la possibilité de jouer contre des IA de différents niveaux.

Enfin j'ai ajouté une sauvegarde des parties pour pouvoir les continuer plus tard [Possible uniquement en mode Humain contre Humain].

Étant fervent d'Easter Eggs je me suis permis d'en insérer quelque un dans le jeu [Il ne vous reste plus qu'à les découvrir].

Tout le codage a été à chaque fois fait dans un fichier « test » d'abord et ensuite inclus dans la version du projet. Je teste aussi des fonctionnalités comme les Macros que je ne connaissais pas .

II. Solutions retenues & difficultés

1 – Difficultés

Toute première difficultés rencontrer celle des macros mais après quelque recherche sur le Web j'ai vite compris comment celle-ci marcher.

Quel temps après une plus grosses difficultés est apparue ; celle de l'IA [comme pour tout le monde je suppose] mais la aussi après beaucoup de recherche sur le Web j'ai réussi a comprendre comment faire " Réfléchir " l'ordinateur.

Enfin j'ai eu quelque difficulté sur les fichiers texte [lecture ; écriture ; rechercher une ligne dedans la récupérer ..] mais après beaucoup de teste et de recherche encore une fois j'ai ressui a bien comprendre leurs fonctionnements , le déplacements etc .

2 – Les solutions

Tout d'abord la grille , pour cette grille je n'ai pas voulu faire une matrice car compliquer a mettre en place .. certes simple d'utilisation avec un simple :
"ma_matrice[ligne][colonne]" pour accéder a la case souhaités.

Je me suis alors demandé si il n'y avait pas un moyen de faire ceci mais avec un simple tableau [pour atteindre le niveau 4 de l'évaluation].

J'ai donc choisis de faire un tableau de CHAR sa taille étant définie par la macro suivante :

```
#define TAILLE_TAB (NB_LIGNE * NB_COLONNE)
```

NB_LIGNE et NB_COLONNE étant des constantes dans le programme.

Ensuite je me suis demandé comment accéder a une case de se tableau.

J'ai donc chercher une correspondance entre le numéro de la case et , la ligne / la colonne.

Ceci m'a donc amené a créer une macro :

```
#define CASE(LIGNE,COLONNE) (LIGNE*NB_COLONNE+COLONNE)
```

Ex : sur une grille de 5 par 2 soit : $5 * 2 = 10$ cases

1	2	3	4	5	
0	1	2	3	4	1
5	6	7	8	9	2

- La case 1 du tableau correspond a CASE(0,0) pour preuve :
 $0 * 5 + 0 = 0$ [donc bien la case 0]
- La case ayant pour chiffre 5 est la 6 eme case : CASE(1,1)
 $1 * 5 + 1 = 6$ [6 eme case du tableau]
- La case ayant pour chiffre 8 est en 9eme position CASE (4,1)
 $1 * 5 + 4 = 5 + 4 = 9$ [donc bien la 9 eme case]

Ensuite L'évaluation de la grille pour se faire plusieurs évaluations sont fait :

- On vérifie une grille passée en paramètre suivant plusieurs points :
 - Si la grille est gagnante pour l'IA on retourne un score très haut
 - Si elle est perdante pour l'IA on retourne un score très bas
 - Sinon

On compte le nombre de groupe de 2 pions alignés par l'IA , le nombre de groupe de 3 pions alignés par l'IA de même pour les pions humains et on retourne la différences des 2 avec des coefficients affecter dans le but de favoriser l'alignement de 3 pions (plus proche de la victoire) a l'alignement de 2 pions , pour se faire on utilise des coefficients.

Cette évaluation est faite dans tous les sens [Diagonales dans le sens '/' ; Diagonales dans le sens '\', Horizontale , Vertical].

L'IA favorisera dans cet ordre :

- Un coup gagnant
- Une défense pour éviter de perdre
- Le plus haut score obtenue par la fonction d'évaluation.
- Sinon elle joue au milieu car c'est le coup qui offre le plus de possibilité pour gagner.

Pour se faire elle utilise l'algorithme MIN-MAX pour améliorer ceci , on utilise l'élagage Alpha-Beta.

La fonction MIN-MAX teste tous les coup possibles et évalue les grilles obtenues.

L'élagage Alpha – Beta quand a lui sert a limiter le nombre de calcule. Grâce a lui on ne calculera pas les sous arbre inintéressant. [Ex : Si jouer dans la colonne 1 est gagnant , jouer dans la colonne 1 et non pas : Calculer pour la colonne 2 jusqu'à la colonne N puis retourner la colonne la mieux ; c'est a dire ici 1]

Pour les sauvegardes de grilles et la gestions des statistiques , j'ai choisis de le faire dans un fichier texte . Ceci aidant a la portabilité et évité des connections a une base de données .

De plus pour afficher les règles j'ai choisis de le faire a partir d'un fichier texte afin de facilité les modifications dedans.

III. Amélioration

La première amélioration qui me vient en tête est (naturellement) celle de transférer le puissance 4 du mode console au mode graphique . Mais n'ayant aucune base dans les bibliothèque graphique et par manque de temps d'apprendre je ne l'ai pas fais , ainsi je le propose en amélioration.

Seconde amélioration ; comme dit précédemment serait de pouvoir supprimer les grilles sauvegarder.

Enfin dernière amélioration auquel j'avais pensé programmer le jeu en réseau pour pouvoir affronter des ami(e)s dans le monde entier .