

Présentation SIM202

Jonas Benhamou, Hugo Blanc, Pierre Cortambert, Jean-michel Mamfoumbi

March 18, 2021

ENSTA Paris

1. Implémentation du jeu d'échecs
2. Implémentation du min-max
3. Validation du min-max et application aux échecs

Implémentation du jeu d'échecs

Définition des objectifs

Implémenter les règles relatives au jeu d'échec (pièces, échiquier, conditions de fin de partie...)

Générer les positions suivantes à partir de la position actuelle

Implémenter l'algorithme du min-max qui permettra à un ordinateur de jouer

Implémenter une heuristique avancée

Implémentation du jeu d'échec

```
Tn  Cn  Fn  Dn  Rn  Fn  Cn  Tn  8
Pn  Pn  Pn  Pn  Pn  Pn  Pn  Pn  7
                                         6
                                         5
                                         4
                                         3
Pb  Pb  Pb  Pb  Pb  Pb  Pb  Pb  2
Tb  Cb  Fb  Db  Rb  Fb  Cb  Tb  1
a   b   c   d   e   f   g   h
Quelle couleur voulez-vous jouer?
```

Génération des positions possibles

On affiche les différents coups possibles

Tn	Cn	Fn	Dn	Rn	Fn	Cn	Tn	8
Pn	Pn	Pn	Pn	Pn	Pn	Pn	Pn	7
								6
								5
								4
								3
Pb	Pb	Cb	Pb	Pb	Pb	Pb	Pb	2
Tb	Tb	Fb	Db	Rb	Fb	Cb	Tb	1
a	b	c	d	e	f	g	h	

Tn	Cn	Fn	Dn	Rn	Fn	Cn	Tn	8
Pn	Pn	Pn	Pn	Pn	Pn	Pn	Pn	7
								6
								5
								4
								3
Pb	Pb	Cb	Pb	Pb	Pb	Pb	Pb	2
Tb	Tb	Fb	Db	Rb	Fb	Cb	Tb	1
a	b	c	d	e	f	g	h	

Tn	Cn	Fn	Dn	Rn	Fn	Cn	Tn	8
Pn	Pn	Pn	Pn	Pn	Pn	Pn	Pn	7
								6
								5
								4
								3
Pb	Pb	Cb	Pb	Pb	Pb	Pb	Pb	2
Tb	Tb	Fb	Db	Rb	Fb	Cb	Tb	1
a	b	c	d	e	f	g	h	

Tn	Cn	Fn	Dn	Rn	Fn	Cn	Tn	8
Pn	Pn	Pn	Pn	Pn	Pn	Pn	Pn	7
								6
								5
								4
								3
Pb	Pb	Cb	Pb	Pb	Pb	Pb	Pb	2
Tb	Tb	Fb	Db	Rb	Fb	Cb	Tb	1
a	b	c	d	e	f	g	h	

Tn	Cn	Fn	Dn	Rn	Fn	Cn	Tn	8
Pn	Pn	Pn	Pn	Pn	Pn	Pn	Pn	7
								6
								5
								4
								3
Pb	Pb	Cb	Pb	Pb	Pb	Pb	Pb	2
Tb	Tb	Fb	Db	Rb	Fb	Cb	Tb	1
a	b	c	d	e	f	g	h	

Tn	Cn	Fn	Dn	Rn	Fn	Cn	Tn	8
Pn	Pn	Pn	Pn	Pn	Pn	Pn	Pn	7
								6
								5
								4
								3
Pb	Pb	Cb	Pb	Pb	Pb	Pb	Pb	2
Tb	Tb	Fb	Db	Rb	Fb	Cb	Tb	1
a	b	c	d	e	f	g	h	

Génération de positions par l'ordinateur en début de partie

Définition de 5 classes :

- une classe **position-echec**

- une classe échiquier

- une classe pièce

- une classe type pièce

- une classe coup

Choix de structures adaptées

Gestion des coups spéciaux

Gestion de la promotion

Robustesse du code

Implémentation du min-max

Arbre des coups possibles

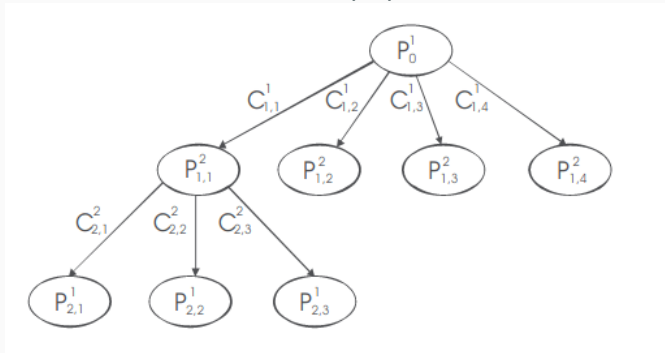
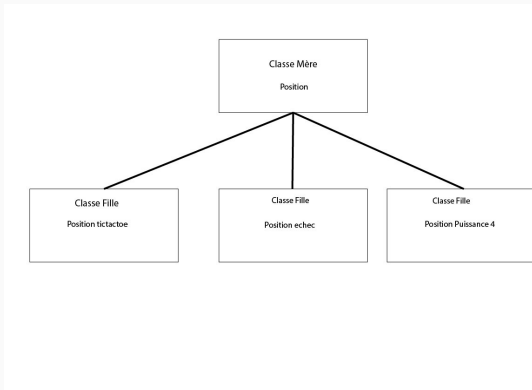


Schéma issu de la présentation du projet

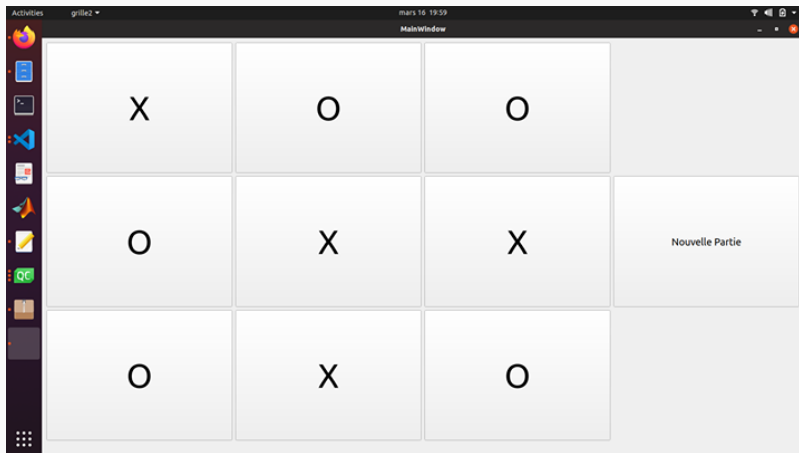
Version générique

L'implémentation avec des classes génériques permet de décliner notre programme pour tout les jeux à deux joueurs qui jouent de façon successive.



Validation du min-max et application aux échecs

Démo : lancement d'une partie de tictactoe



Interface graphique du tictactoe

Définition de 2 classes :

- une classe **grille** qui stocke les coups

- une classe **position-tictactoe** qui hérite de position

Fonctions utiles :

- les fonctions **valeur** et **genererposition** nécessaires au tictactoe

- les fonctions vérifiant l'état du jeu (en cours, terminé, victoire d'un joueur, nul...)

- la fonction d'affichage de la partie en cours

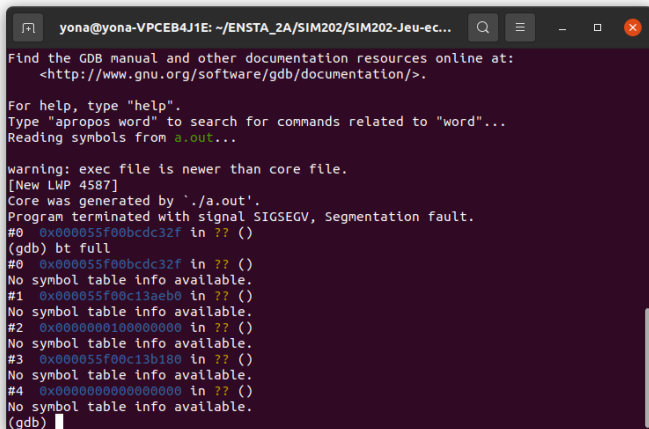
Problèmes liés à l'héritage de la classe **position** qui est abstraite

Travail avec beaucoup de pointeurs différents, nécessité de vérifier que les pointeurs sont non nuls

Nécessité de transtyper certains objets selon les fonctions qu'on utilisait

Problèmes rencontrés

Il faut tester si un pointeur est nul !



```
yona@yona-VPCEB4J1E: ~/ENSTA_2A/SIM202/SIM202-Jeu-ec...
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from a.out...

warning: exec file is newer than core file.
[New LWP 4587]
Core was generated by `./a.out'.
Program terminated with signal SIGSEGV, Segmentation fault.
#0  0x000055f00bcd32f in ?? ()
(gdb) bt full
#0  0x000055f00bcd32f in ?? ()
No symbol table info available.
#1  0x000055f00c13aeb0 in ?? ()
No symbol table info available.
#2  0x0000000010000000 in ?? ()
No symbol table info available.
#3  0x000055f00c13b180 in ?? ()
No symbol table info available.
#4  0x0000000000000000 in ?? ()
No symbol table info available.
(gdb)
```

Utilisation du mode debug.

Pour faire l'héritage il faut disposer des fonctions:

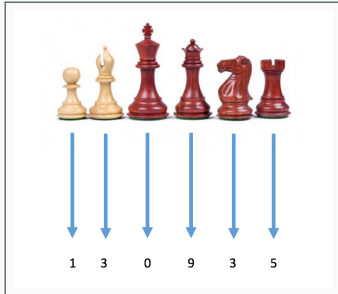
-**genererpositionfils()**

-**valeur()**

Pour coder la fonction **genererpositionfils()**, on code pour chaque pièce une fonction qui permet de savoir quelles sont les cases que la pièce peut atteindre.

Pour la fonction **valeur** il faut définir une heuristique.

Choix de l'heuristique



Attribution des points

Différentes fonctions valeur :
Soit N le nombre de pièces en jeu,

$$Val_1 = \sum_{pieces} V_{piece} * (N_{joueur} - N_{adv})$$

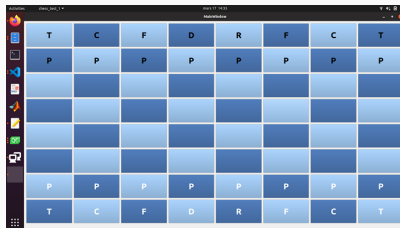
où $V_{piece} = cte$

$$Val_2 = Val_1 + \beta (C_{joueur} - C_{adversaire})$$

Pistes d'améliorations pour l'heuristique.

- Définir un autre poids pour les pièces.
- Pénaliser certaines positions : 8 pions sur la même ligne, plusieurs pions sur la même colonne...
- Favoriser certains coups : Avancement d'un pion pour être promu
- Travailler sur les coefficients α et β

Résultats et démonstration



Interface graphique du jeu d'échec

Merci de votre attention.
