Réalisation d'un jeu d'échec modifié Stackchess

Mame Diarra Touré

Le 26 décembre 2018

ENSIIE
École nationale supérieure d'informatique pour l'industrie et l'entreprise
1 Square de la Résistance
91025 Evry cedex
https ://www.ensiie.fr/

Sommaire

1 Introduction

Le but de ce projet est d'implanter un jeu d'échecs modifiés,Stackchess.
Dans cette version modifiée du jeu d'échecs, il est possible d'empiler les pièces d'une même couleur sur une même case. À chaque tour, un joueur peut déplacer un nombre quelconque de pièces situées sur le haut de la pile d'une case donnée à condition qu'il s'agisse de ses pieces .
Néanmoins, le déplacement doit être légal pour chacune de pièces déplacées. Par ailleurs, il ne suffit pas de se déplacer sur une case contenant une pièce adverse pour la prendre ; il faut déplacer un nombre de pièces strictement supérieur à celui des pièces adverses dans la case atteinte. Le jeu a été implémenté en langage C.

2 L'échiquier

J'ai, initialement, voulu représenter l'échiquier comme une matrice de piles de tableaux de chaines de caractères (char) et donc avec une structure en 4 dimensions. Cependant j'ai rencontré des difficultés dans la mise en place et la manipulation de la structure de piles de tableaux de char notamment pour l'empilement et le dépilement ainsi que l'allocation mémoire.
J'ai donc décidé d'utiliser une matrice de piles d'entiers (structure en 3D) et de mettre en place une fonction qui allait faire l'équivalence entre les entiers et les noms des pièces. Une fois la structure mise en place, je me suis occupée de l'affichage.
L'affichage n'a pas posé de problème hormis le fait que les éléments de la case sélectionnée doivent s'afficher à droite de l'échiquier et qui dans notre cas s'affichent en bas. J'ai utilisé la table ASCII pour pouvoir afficher les lettres a, b, c ,... représentant les noms des colonnes .

3 Les déplacements

Le plus gros du travail a été l'implémentation des déplacements. Au départ j'avais pensé distinguer les pièces noires des pièces blanches et de traiter les différents cas de figure. Je me suis rendu compte qu'à part les pions (PN et PB) pour lesquelles la position de départ importait, les autres pièces avaient les mêmes degrés de liberté qu'elles soient blanches ou noires.
J'ai donc implémenté d'une façon général les déplacements pour les tours, les dames, les rois et les fous et distinguer le cas pions noires et pions blancs. Une fois la condition sur le sens de déplacement vérifié (vertical, horizontal et/ou en diagonal) a l'aide de la fonction estlegalpiece, il a fallu vérifier que le joueur ne sautait pas au dessus d'une case non vide. J'avais initialement inclus cette vérification dans la fonction estlegalpiece mais cela entrainait une imbrication de plusieurs boucles for. J'ai donc par soucis d'optimalité préféré faire cette vérification à part. Cela permettait également de faire la vérification qu'une seule fois lorsqu'on désirait déplacer plusieurs pièces d'une case donnée en même temps. Une fois assuré de la légalité, j'ai implémenté la fonction qui se charge du déplacement à proprement parler c'est-à-dire la fonction qui dépile de la case de départ le nombre de pièces désiré et qui les empile sur la case d'arrivée en conservant bien le sens d'empilement.
Ce dernier point a nécessité l'utilisation d'un tableau intermédiaire où sont depilés dans un premiers temp les éléments de la pile de départ pour ensuite être empiler dans la pile d'arrivée en commençant par le dernier élément du tableau.

4 le jeu

Chaque joueur est invité à tour de rôle à sélectionner une case de l'échiquier, les colonnes étant nommées a, b, c, ... et les lignes 1, 2, 3, ... la fonction strtol m'a permis de recueillir le choix du joueur sous la forme a6, b5, c4 etc.... et d'en déduire la ligne et la colonne correspondant.
La partie se termine lorsqu'il ne reste plus que des pièces de la même couleur.

5 Conclusion

Mon programme fonctionne de manière très satisfaisante pour un tableau de dimension 66 mais certaines fonctions comme InitialisationJeu par exemple ne sont pas assez automatisée pour permettre l'extension du tableau en une taille supérieure. De plus dans la fonction principale je n'ai pas réalisé de vérification sur les données entrées par les joueurs c'est-à-dire s'ils sont ou non dans le format adéquat ce qui est problématique lorsqu'un joueur entrer un format inadéquat lors de ses choix. En outre le cas d'égalité (match nul) n'est pas pris en charge par le programme.
Enfin, il aurait été sans doute plus judicieux de ma part de nommer les piles dans InitialisationJeu en fonction de leur position et non pas de leur contenu en début de jeu.