CRAZY-Chess

Final Project Report

By

[405262180]

[劉育騏]

[405262049]

[陳銘澤]

X86 Assembly Language

Spring 2018

Date Submitted: January 6, 2018

Abstract:

目的:我覺得相對其他題目是比較有趣的，再加上做遊戲的選擇比較廣，自由度比較高，不會受到限制。結果: 當然是成功的，雖然發生了很多事，以及方法使用較繁雜，但最後還是做出來了，完成度算是100%。那關於這個遊戲玩法，以及寫法，將在下面說明。

Table of Contents:

1.Introduction

2.Project plan

3.Discussion

4.Conclusion

5.Team Work Arrangement

1. Introduction(介紹):

此遊戲是一種棋盤式遊戲，類似象棋+西洋棋，只是棋子功能不大相同，棋子走法遊戲內有說明。遊戲規則:先吃到王的那方，即便勝利。1次只能動一顆棋子。

1. Project plan (專案計畫):

我們想寫出，棋子能夠吃(不能吃自己)、能夠走(並判斷是否是棋子的走法)，並能判斷輸贏、能限制使用者輸入座標格式(並判斷是否符合輸入)、能有先後順序。棋子不能穿透(除了獅子、豹)，且能漂亮的印出棋子，以及遊戲重新開始選擇。

1. Discussion(討論):

關於寫法的部分，我用了1個非常非常爛的方法去儲存這個棋子以及運作，因為用了4個1維陣列，存取X與Y座標，看起來像是8X8的矩陣，但實際上沒印東西的都是空的。如果我是用struct+2維陣列去處理的話，那一切根本不需要寫到1千多行程式碼，有時間的話在大改。至於之前上台的bug穿透，以及邊界判斷都已經修改完成。穿透主要是處理了馬跟車。我把code加了一些條件搜尋。時間複雜度大概是O(N\*N\*(N-1))。怎麼解決的?首先是這樣，因為我條件判斷，原本是這樣寫的:使用者輸入的終點，XY座標，看這XY座標去搜尋自己兩個一維陣列XY，檢查是否有自己人，有的話則不能吃。這樣嚴重的問題是路徑之內有人的話會穿透。於是，我做了，每一格都去搜尋是否存在自己人以及敵人，但最後一格不判斷。於是就這樣完成了。非常麻煩。

我剛說的用2維陣列+ struct的概念是這樣:總共是24顆棋子，每個棋子存XY與棋子的編號。然後用8X8陣列去存。空的陣列放-1那路徑搜尋就變簡單了。只要看是否為那區間是否存在-1就好了。

1. Conclusion(結論):

做這個專題，很花時間，很燒腦袋。再加上組員其實沒有很認真注意組合語言，都到快期限，才開始想。然而我早就做的差不多，雖然方法爛，如果能知道更多觀念，就會很好寫。以後可能會先發想完，而不是邊寫邊想，我決定以後寫專題之前，可能畫個流程圖之類的。這次也是很好玩的經驗

5.Team Work Arrangement

405262180 劉育騏80% 主要code + debug+ think + 書面報告+口頭報告

405262049 陳銘澤 20% 上顏色+註解+變數名稱的調整+debug