Coursework Report

kyle benning
40203951@napier.ac.uk
Edinburgh Napier University - Algorithms and Data Structures (SET09117)

1 Introduction

The task was to implement a checkers game in a language of your own choice, the language choice made being python. The program should feature a form of user interface wither it's text based or uses some form of Graphical User Interface(GUI) and include the use of algorithms to create an AI player.

2 Design

The program included the use of a data structure known as a list to create a game board that was text base and used the characters r,b,R and B to represent regular and kinged pieces as well as underscores and vertical lines to represent the board. The software made use of multiple function to create the game rules such as checks to make sure that the user does not select the opponents pieces or selects a blank space, functions were also used to make pieces automatically jump enemy pieces. The software also including various comments throughout to explain the function of various parts of the code to anyone viewing the code to ensure that they can understand what each section does.

3 Enhancements

If more time was available features that would be added would include the use of AI to allow the user to play against the computer if a second player was not available which would allow the game to be single player rather than always being a two player game. Other features that could have been included would be an undo and redo function so that accidental moves can be taken back or for the undo to be reverted, the inclusion of game history would have also improved the overall quality of the game since the user could review games. A graphical user interface would make the software more appealing overall and would enhance the user experience.

4 Critical Evaluation

Features of the software that worked well would be the user interaction as the game effectively prompts the user on making a move and informs them when a move they have attempted to make are invalid which prevents the user from being confused when taking turns. features that could be improved would include the user interface, which was very basic and unappealing. This would cause the program to have a sense of being cheap and poorly designed. This could be fixed by taking the time to create and make use of a Graphical User Interface which would make the software look not only professional but also of a higher quality.

5 Personal Evaluation

Overall my performance was good when overcoming particular challenges such as having trouble stopping the user from selecting pieces that were not theirs. However my performance could be improved a fair amount by taking the time to add additional features to the software to stop it from being very basic.