 Created by Åse Jevinger

Assignment 1: Othello

The purpose of the assignment is to give a deeper understanding of the alpha-beta pruning search algorithm and how it can be implemented. The purpose is also to give an insight to heuristic evaluation functions.

Preparation:
Read chap. 5 in the book *Artificial intelligence: a modern approach*. Focus particularly on chap. 5.2-5.4. Try to understand how the algorithm in Figure 5.7 works and how it relates to the explaining text.

Simplified rules of the Othello game:
Othello is a game of strategy played by two players. The players take turns placing game pieces, called disks, on a board. The disks are white on one side and black on the other. One player is assigned to white and the other to black. During a turn, the player must place one disk with their assigned color facing up, on the board. The board is divided into squares. The player places the disk in an unoccupied square, and can thereafter turn all disks of the opponent's color that are between the placed disk and other disks of the player's color. The disks are turned in straight lines in any direction. If the disk is not placed next to a board edge, there are 8 possible directions. A placed disk may never be moved. The winner of the game is (usually) the player who has the most disks in his/her own color on the board, when the board is full. For more information about the rules of the game, please see http://www.othello.nu/ or https://en.wikipedia.org/wiki/Reversi.

In this assignment we use a simplified version of the game, meaning that the board only has 4x4 squares (usually 8x8 square boards are used). Furthermore, the players are allowed to place their disk on any unoccupied square and the game always ends when the board is full, i.e. when all 16 squares are occupied (and only then). The first disks may also be placed anywhere on the board.

Assignment:
Implement the Othello game, where a human user plays against the computer. Write an assignment report. The assignment also includes an oral examination.

Game requirements:

- The computer algorithm for finding the best choice of where to place the next disk shall be based on alpha-beta pruning.
- The alpha-beta search shall be cut off based on some criterion. For instance, the search may be cut off after a specified number of draws forward have been examined (i.e. when the tree depth has increased with a specified number) or when a tree leaf is reached.
- A heuristic evaluation function shall be used to estimate the utility of a non-leaf node. For instance, the evaluation function may calculate the number of disks of the computer's color.
- A computer draw shall not take longer than 5 seconds.
- The code should be written in Java.
- The game should be based on the simplified rules above (more advanced rules are also allowed).

- The GUI may be fairly simple. The command prompt can be used for user input and output. For instance, a matrix showing the current status of the board is sufficient.
- The following shall be printed after each draw:
  - The current status of the board
  - The depth of the search
  - How many nodes were examined
- You are allowed to work individually or in groups of two.

Report requirements:

- The expected size of the report is around 5 pages.
- The report shall answer the following questions:
  - How did you build your search algorithm?
  - What evaluation function and cut off criterion did you use?
  - What would you have done differently if you had done it all over again (given the stated requirements)?
  - What experiences have you acquired?
- The report must include a reference list if it refers to external sources.
- Naturally, the report shall be written in your own words (i.e. slightly modified or simply translated texts written by someone else, will be rejected).
- The report shall be well-structured with relevant headings and correct paragraph groupings.
- The report shall be well-written and proofread several times. It must be written with correct spelling and sentence construction.

Submission:
The final report shall be a word or pdf document. The report and the source code should be submitted as a zip-file on the assignment submission page on Itslearning (one submission per group is sufficient). This assignment has three submission deadlines, which are specified on the assignment submission page on Itslearning. If you, after your submission, need to make minor changes to your assignment, you are allowed to make one resubmission. However, if you need to make major changes, if your assignment is not approved after one resubmission, or if you miss the submission deadline, you will have to (re)submit using one of the submission deadlines later in the course. If your assignment is not approved after the final opportunity to submit, you need to come back the next time the course is given. After the submission has been approved, you should meet the teacher responsible for the assignment for a short discussion about your assignment (the teacher will announce the time for the discussion). In order to pass this assignment, you need to pass both the report part and the discussion part of the examination.