# Generate next move flow v1

Minimax without alpha-beta pruning

Parameters: deptToGo

Idea: search (expanding a node) in very important in the performance of the game, thus you should take it in consideration. You can use a strategy that expands the best node first. Note: this ordering can be acheived when generating all possible moves. This way, you will add in tree the moves ordered by how best they are. The “best” term is heuristical: it is probably not the real best move, but the better the ordering, the better the performance will be. Second note: if the ordering would be perfect (this is not posible in real world) the searching algorithm would not be needed anymore.

Another idea: after the black possible moves were generated, flip the board, so that the same move generation rules caan be applied to white moves. (pown goes one step up, but the opponent’s pawn goes down if you do not rotate de board.)

While(depth<depthToGo)

blackPossibleMoves=generateAllPossibleMoves(black)

tree.add(blackPossibleMoves) // adds all possible moves as children for the current whiteMove.

Foreach(currentBlackMove : blackPossibleMoves)

whitePossibleMoves=generateAllPossibleMoves(white)

tree.add(whitePossibleMoves) //adds all possible moves as children for the currentBlackMove.

Depth++;

Foreach(leave: leaves)

evaluationFunction(leave)

pick move sequence with the highest evaluation function value.

Return the corresponding black move