**CSC384 Assignment 2 Design: Checkers**

**Calculating Utility of Terminal States**

The utility of a terminal state is either +infinity for the winning player or -infinity for the losing player. This is calculated by iterating through every character in the board; if the given player is red (it is red’s turn), for example, and there are no black pieces (including kings) present, the function returns +infinity since the red player has won. This operates similarly for the black player.

Furthermore, if neither condition of having an entire player wiped out is met, the generate successors function is called for the whichever player’s turn it is. If it returns an empty list, this means that there are no possible moves left; thus, the utility function would return -infinity for this losing player. There is a final “else” statement for error-checking – if this runs, it means that the game is not yet over, and we have not actually reached a terminal state yet. In this case, the evaluation function should instead have been used to estimate the player’s utility for that state.

**Estimating Utility of Non-Terminal States: Evaluation Function**

For non-terminal states, an evaluation function was used to estimate a player’s utility. The search for terminal states stops at a specified depth limit, which was \_\_\_ in this case. Once the depth limit was met, the evaluation was called if it was a non-terminal state.

The evaluation function for a given player was calculated by iterating through every piece on the board. For each player, a regular piece counted as 1 point, while a king counted as 2 points. The final utility estimate for the player was found by subtracting the opponent’s point total from the given player’s point total for that state.

* Talk about points for each type of player → give example if extra space?
* Expand on how optimized (ex. more points for better locations, etc.) → add later!

**Additional Optimizations**