CS 201 Spring 2019 Portfolio Project Proposal

Project Choice: Option 2 – Custom Game – Othello

**Rules and Specifics of the Game**

* The board, traditionally green, is an 8x8 grid.
* There are two players, one with black pieces and one with white pieces.
* The game begins with two pieces of each color diagonal from each other in the center four grid locations, as shown below.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  | w | b |  |  |  |
|  |  |  | b | w |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

* The player with black pieces moves first.
* The player with black pieces must place a black piece on the board such that there is a horizontal, vertical, or diagonal line between the new piece and an existing black piece, and there must be one or more contiguous white pieces in between the two black pieces. Legal moves for a black piece are denoted with asterisks below.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  | \* |  |  |  |  |
|  |  | \* | w | b |  |  |  |
|  |  |  | b | w | \* |  |  |
|  |  |  |  | \* |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

* The moves for a black piece denoted by an asterisk and shown below are not legal because there are no white pieces between the black pieces, or the white pieces are not contiguous.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  | w | b |  |  |  |
|  |  |  | b | w |  |  |  |
|  |  |  | \* |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  | \* |  |  |
|  |  |  | w | b |  |  |  |
|  |  |  | b | w |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | \* |  |  |  |  |
|  |  |  | w |  |  |  |  |
|  |  |  |  | b |  |  |  |
|  |  |  | w | w |  |  |  |
|  |  |  | b | w |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| \* |  |  | w | b |  |  |  |
|  |  |  | b | w |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

* After the black piece is placed, all white pieces in between the two black pieces are “flipped” to white. A before and after example is shown below, with the move and effected piece indicated by asterisks.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  | \* |  |  |  |  |
|  |  |  | w | b |  |  |  |
|  |  |  | b | w |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  | b |  |  |  |  |
|  |  |  | b\* | b |  |  |  |
|  |  |  | b | w |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

* Now the turn of the player with the black pieces. It is now the turn of the player with the white pieces.
* The player with the white pieces makes move similarly to the black pieces. The player with white pieces must place a white piece on the board such that there is a horizontal, vertical, or diagonal line between the new piece and an existing white piece, and there must be one or more contiguous black pieces in between the two white pieces.
* Players continue making moves back and forth in this way.
* If a player cannot make a move, the player’s turn is skipped.
* If both players cannot make a move, the game is over, regardless of if the board is filled.
* If the board is filled, the game is over.
* Once the game is over, the winner is the player with the most pieces of their color on the board.
* It is possible for there to be equal numbers of black and white pieces at the end of the game. This is considered a tie, and the game is still over.

**Activities of the Program**

What will I need to make this program do to successfully play this game?

* Display
  + menu options, board, scores, instructions, etc.
* User Input
  + Read user input
  + Check input validity
  + React appropriately
* Gameplay
  + Check to see if game is over
    - Are there any valid moves left for the player whose turn it is?
    - If not, are there any valid moves left for the other player?
    - Either skip turn or end game.
  + Place piece on the board
    - Add new piece
    - Flip any pieces that need to be flipped by addition of new piece
  + Keep track of score
    - Keep track of wins across consecutive games
    - Keep track of current score, the number of pieces each player has on the board
  + Easy Mode Against Computer
    - For each valid move, calculate the number of pieces that would be flipped
  + Regular Mode Against Computer
    - For each valid move, compare the value assigned to that square to find the highest value move.

**Concept: User Interface Design**

* ASCII interface
* Primary Menu

OTHELLO

Welcome to Othello!

At any point, enter “main” to return to this screen.

Enter “next” to move to the menu.

* + Input Options (persist throughout program):
    - main – display primary menu
    - back – display previous menu
    - \*any invalid input\* - display error message containing list of valid inputs for that menu
  + Input Options (this menu):
    - next – move to secondary menu
  + Invalid Input Display

OTHELLO

That input is invalid.

Enter “next” to proceed to the menu.

* Secondary Menu

OTHELLO

Please choose game mode.

Enter “double” to start a two-player game.

Enter “computer” to play as a single player against the computer.

* + Input Options:
    - double – display gameplay instructions
    - computer – display computer menu
  + Invalid Input Display

OTHELLO

That input is invalid.

Enter “main” to return to the home screen.

Enter “double” to start a two-player game.

Enter “computer” to play as a single player against the computer.

* Computer Menu

OTHELLO

You have selected to play against the computer.

Please select a level of difficulty.

Enter “easy” or “regular”.

* + Input Options:
    - easy – display gameplay instructions
    - regular – display gameplay instructions
  + Invalid Input Display

OTHELLO

That input is invalid.

Enter “main” to return to the home screen.

Enter “easy” or “regular” to select a level of difficulty.

* Gameplay Instructions

OTHELLO

The board will be displayed as below.

1 \* \* \* \* \* \* \* \*

2 \* \* \* \* \* \* \* \*

3 \* \* \* \* \* \* \* \*

4 \* \* \* w b \* \* \*

5 \* \* \* b w \* \* \*

6 \* \* \* \* \* \* \* \*

7 \* \* \* \* \* \* \* \*

8 \* \* \* \* \* \* \* \*

a b c d e f g h

The numbers and letters along the side of the grid are used as identifiers for rows and columns.

An asterisk \* represents any empty square.

A w represents a square with a white piece played on it.

A b represents a square with a black piece played on it.

Players take turns placing pieces, with black going first.

A piece of one color must be placed on the board such that there is a horizontal, vertical, or diagonal line between the new piece and an existing piece of the same color, and there must be one or more contiguous pieces of the opposite color between these two pieces.

To place a piece, enter the coordinates of a valid square. For example, to place a black piece in the sample board above in the square above the top-left white piece, enter “d3”.

Enter “next” to start gameplay.

* + Input Options:
    - next – display gameplay for mode selected earlier
  + Invalid Input Display

OTHELLO

That input is invalid.

Enter “main” to return to the home screen.

Enter “next” to start gameplay.

* Gameplay

OTHELLO

Wins: Player One – 0 Player Two – 0

Score: black – 2 white - 2

Current Turn: Player One (black)

1 \* \* \* \* \* \* \* \*

2 \* \* \* \* \* \* \* \*

3 \* \* \* \* \* \* \* \*

4 \* \* \* w b \* \* \*

5 \* \* \* b w \* \* \*

6 \* \* \* \* \* \* \* \*

7 \* \* \* \* \* \* \* \*

8 \* \* \* \* \* \* \* \*

a b c d e f g h

* + Input Options:
    - \*coordinates\* - redisplay gameplay, with changes made from placing new piece
  + Invalid Input Display – coordinates indicate a space that is not a valid move

OTHELLO

That is not a valid move.

A piece of one color must be placed on the board such that there is a horizontal, vertical, or diagonal line between the new piece and an existing piece of the same color, and there must be one or more contiguous pieces of the opposite color between these two pieces.

Enter “main” to return to the home screen.

Enter coordinates of a valid square to place a piece.

* + Invalid Input Display – not coordinates

OTHELLO

That input is invalid.

Enter “main” to return to the home screen.

To place a piece and take your turn, enter coordinates, like “d3” or “g8”.

**Concept: Play-Against-Computer AI**

* After selecting to play against the computer, the user can choose between “easy” or “regular” gameplay modes.
* Easy – Maximum Disc Strategy
  + This strategy is the most intuitive and mimics how a new player would play the game. The strategy is simple: make the move that flips the maximum number of disks.
  + However, this method of gameplay is ineffective. Flipping more disks, especially early in the game, simply offers your opponent more options for strategic gameplay. Also, disks flipped early in the game and in the center of the board are not stable, and therefore do not matter.
  + Providing an easy mode of gameplay allows a new user to experiment with different strategies.
* Regular Gameplay – Weighted Square Strategy
  + This strategy recognizes that certain squares are more strategically valuable than other squares, so each square has a value assigned to it. The different types of squares, in order from least valuable (worst) to most valuable (best) are:
    1. X-Squares – squares diagonally adjacent to a corner
    2. C-Squares – squares vertically or horizontally adjacent to a corner
    3. Danger Zone – squares in the rows just inside the edge rows
    4. Internal Squares – squares in the center of the board
    5. Internal Edges – center two squares of each edge
    6. Internal Corners – squares diagonally adjacent to the X-squares
    7. Edges – squares between the internal edge squares and C-squares
    8. Corners – four corner squares, the most desirable in the board

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 8 | 2 | 7 | 5 | 5 | 7 | 2 | 8 |
| 2 | 1 | 3 | 3 | 3 | 3 | 1 | 2 |
| 7 | 3 | 6 | 4 | 4 | 6 | 3 | 7 |
| 5 | 3 | 4 | 4 | 4 | 4 | 3 | 5 |
| 5 | 3 | 4 | 4 | 4 | 4 | 3 | 5 |
| 7 | 3 | 6 | 4 | 4 | 6 | 3 | 7 |
| 2 | 1 | 3 | 3 | 3 | 3 | 1 | 2 |
| 8 | 2 | 7 | 5 | 5 | 7 | 2 | 8 |

* + Of all squares available to play on, the square with the largest value assigned to it is chosen.

**Implementation: Check for Valid Moves**

This action must be performed at the beginning of every turn to determine if a turn should be skipped or if the game is over. Stores a list of coordinates of valid moves in an array. These coordinates are used when determining the next move for AI, and for determining if a user move is valid.

for every piece on the board

if the piece is the color of the turn (black turn, black piece)

search above for empty square, stop if you hit a piece of the same color

search above-right for empty square, stop if you hit a piece of the same color

search right for empty square, stop if you hit a piece of the same color

search below-right for empty square, stop if you hit a piece of the same color

search below for empty square, stop if you hit a piece of the same color

search below-left for empty square, stop if you hit a piece of the same color

search left for empty square, stop if you hit a piece of the same color

search above left for empty square, stop if you hit a piece of the same color

**Implementation: Flip Pieces**

**Questions and Clarifications on Instructions**

* Othello must be played by two people, but the instructions for the portfolio specify that “single” must be an option for game mode. For this game, can the “single” game mode be omitted so that the only options are “double” and “playing against the computer”?
* How intelligent should the AI for “playing against the computer” be? I detail my suggested AI for “easy” and “regular” gameplay mode in this proposal, but Othello computers exist that can beat even the most dedicated players through game trees and the Minimax algorithm. Are my current AI goals sufficient for this project, or should I pursue a more ambitious computer mode?
* The other games all are of “arbitrary board size”. A standard Othello board size is an 8x8 grid. Should there be an option for the user to select a board of a different size?