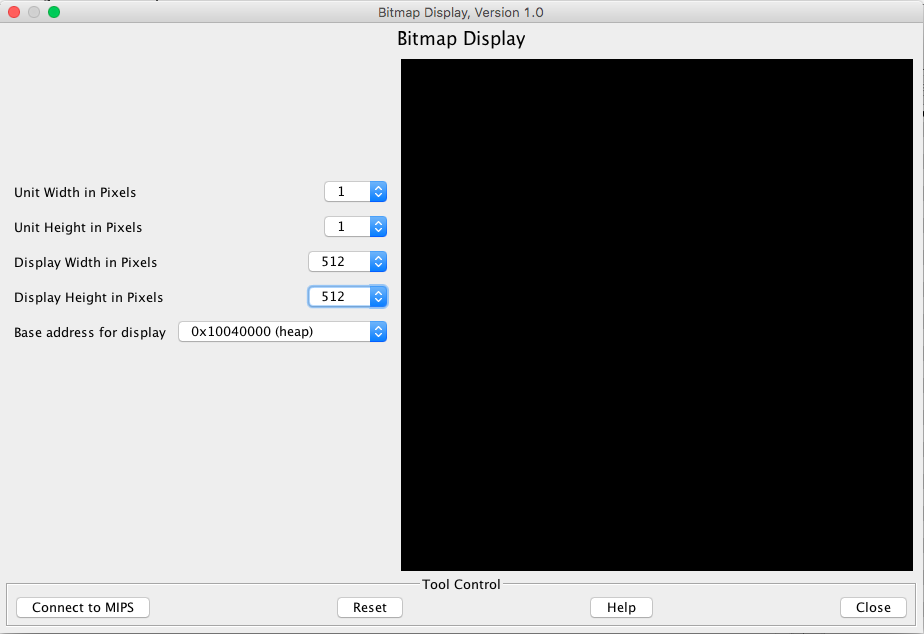
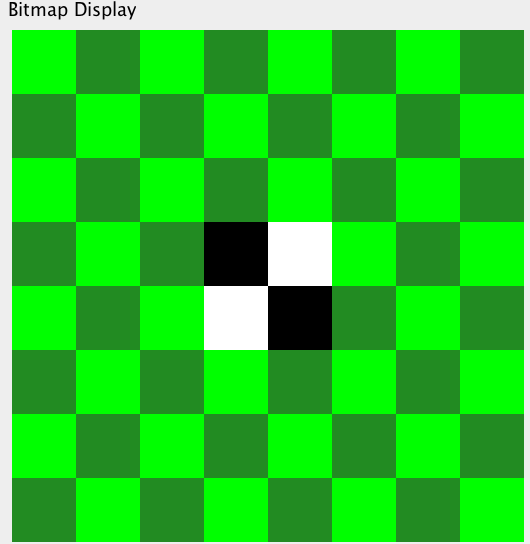
Instructions to play Reversi:

1. Place MARS v4.5 executable into an empty directory.
2. Place all supporting files in the same folder as the MARS v4.5 on your local machine. The required source files are main.asm and validateAI.asm, BitmapMenu.asm, BitmapWin.asm and BitmapLost.asm. Also place the bitmap images you\_win.bmp, you\_lose.bmp, and reversi\_the\_game.bmp into the folder.
3. Execute MARS v4.5
4. Open main.asm on MARS.
5. Under “Settings,” check the “Assemble all files in directory” option
6. Under “Tools,” open the Bitmap Display
7. Set both the “Unit Width in Pixels” and “Unit Height in Pixels” to 1. Set both the “Display Width in Pixels” and “Display Height in Pixels” to 512. Select “heap”as the base address for display. See below. 
8. Press the “Connect to MIPS” button.
9. Assemble the program by pressing the F3 button.
10. Run the program by pressing the F5 button.
11. The console will prompt the user to enter ‘1’ to play Reversi or ‘2’ to exit the program. Enter 1.
12. The game board will appear such that light and dark green squares are empty spaces, black spaces are occupied by the user, and white spaces are occupied by the computer player. See below.
13. The console will first prompt the user to enter the column number (x position) for their move. The number must between 1 and 8, inclusive, with 1 being the leftmost column and 8 being the rightmost column.
14. The console will then prompt the user to enter the row number (y position) for their move. The number must between 1 and 8, inclusive, with 1 being the top row and 8 being the bottom row.
15. After the user has entered the row and column for their move, the console will report whether the user entered an invalid move. A move is considered invalid if
    1. the input location does not sandwich the opponent’s pieces in the up-down, left-right, or diagonal direction,
    2. the input location is already occupied,
    3. or the input location is beyond the bounds of the board.

If the user enters an invalid location, the console will prompt the user again for a row and column. Note: The program will only prompt the user for a move if a valid location is available on the board.

1. After the user has entered a valid move, the computer player will place their move. Note: The computer player will take multiple turns if there is no valid move for the user.
2. After the computer player takes their turn, the score of the user (black) and computer player (white) will be displayed. The user will then be prompted again after the computer player takes their turn.
3. Repeat steps 13-17 until the end game condition is met. The game ends when there are no longer any valid moves for either the user or the computer player.
4. The player with the higher score at end game is the winner and the other player is the loser.
5. The program will stop running after the winner is declared. To replay the game immediately, go to step 9.