CS 2340.006

Project: Tic-Tac-Toe Using MIPS Assembly Language

**Submitted by:**

Trisha Espana

Jahnzeb Bastaki

***Submitted to:***

Professor Wafa Jaffal

01 May 2023

1. **PROJECT REPORT**
2. **Description**

The program created has a goal of replicating the famous classic game of “Tic-Tac-Toe”. Using MARS as our Interactive Development Environment (IDE) and MIPS Assembly Language. The said game is played by two (2) players and for this project, a human user will have the computer as his/her opponent. The Tic-Tac-Toe game is modeled with the use of a 3 by 3 grid and ‘X’s and ‘O’s are the most commonly used characters. The objective of the game is to win by having a consecutive pattern, either horizontally, vertically or diagonally without the opponent disrupting the pattern.   
This program is built in a way wherein the players get to take turns in putting their assigned character onto the grid. The computer will not necessarily analyze how to win, rather, it is programmed to pick an available or empty slot on the board. The winner is set to be determined after a player’s third move. When no winner is established, a tie will be concluded.

1. **Challenges Encountered**
2. **Things Learned**
3. **Algorithms and Techniques Used in the Program**
4. **Contributions**
5. **Suggestions (Optional)**

**2. VIDEO CLIP LINK**

**3. MIPS ASSEMBLY LANGUAGE MODULES**

**4. USER MANUAL**

Welcome to user manual for the Tic-Tac-Toe game implemented in MIPS Assembly language by our group. This manual will go over the rules of the game, how to play the game, and how to run the MIPS program on your local machine.

**Introduction**

Tic-Tac-Toe is a two-player game played on a 3 x 3 grid where players take turn in marking spaces with symbols (Usually X and O) until either player achieves the win condition or the grid is used up and ends in a draw. In this program the user will be playing against the computer in a Tic-Tac-Toe game.

**MIPS Assembly Environment Setup**

Before running the game, ensure that you have a MIPS Assembly Environment set up on your local machine. If you do not have a MIPS Assembly Environment, you can download one here at: <https://courses.missouristate.edu/kenvollmar/mars/download.htm>.

We will be using this MIPS Assembly Environment to run the game

**Run the Game**

Once the MIPS Assembly Environment is established, to run the Tic-Tac-Toe game, perform the following:

1) Open the MIPS Assembly Environment

2) Go to the File drop down menu and locate Open.

3) Browse through your file directory to locate the tic-tac-toe.asm file that is provided and open that file.

4) Go to the File drop down menu and Save the file.

5) Go to the Run drop down menu and click on Assemble.

6) Click the green Run button in the menu to run the game.

Logo

Description automatically generated

**Game Controls**

The game utilizes a simple text-based interface where players enter their moves by specifying the row and column number of the grid space that is desired. To make a move, enter the number for the space on your turn. The grid is numbered as follows:

1 | 2 | 3

---+---+---

4 | 5 | 6

---+---+---

7 | 8 | 9

**Gameplay**

The game follows these rules:

1) First the program asks the Player whether the players want to play as an ‘X’ symbol or an ‘O’ symbol. If the Player picks ‘X’ then the Player goes first, but if the Player picks ‘O’ then the Computer goes first.

2) The Player and the Computer then take turns in making their move.

3) To make a move, enter a number corresponding to the desired empty space on the grid.

4) The game will then validate the whether the number picked is between 1 and 9 and will also validate whether the space is empty or not.

5) The first player that satisfies the win condition wins the game.

6) A winning condition happens when the player or computer has three of their respective symbols in a row, column, or in a diagonal and a win or lose message will appear.

7) If all the spaces are filled and the win condition is not met, the game will display a message that will indicate that the game has ended in a tie.

8) Once the game reaches a tie or a win condition, the game will terminate.

**Conclusion**

This marks the end of the user manual. You are now familiar as to how to run and play the game of Tic-Tac-Toe created by our group in MIPS Assembly Language.

**References**

https://www.surfertoday.com/games/tic-tac-toe

https://courses.missouristate.edu/kenvollmar/mars/download.htm