
Implementation of a Tic-Tac-Toe game Using Python Environment for gaming application

¹Chitaraj Karan (USN : 1DS21EC058), ¹Harsh Raj Karan (USN : 1DS21EC074),

¹Abhijith Jayarajendra Nayak (USN : 1DS21EC004), ¹Khushi (USN : 1DS21EC095),

²Dr. H.V. Manjunath, ³Adithya T.G., ⁴Dr. Pavithra G., ⁵Dr. Sindhu Sree M.,

⁶Dr. T.C.Manjunath* Ph.D. (IIT Bombay), Sr. Member IEEE, Fellow IE, Chartered Engineer

¹First year BE UG (ECE) Second Sem Students, Dept. of Electronics & Communication Engg.,

Dayananda Sagar College of Engineering, Bangalore, Karnataka

²Professor & mini-project guide, ECE Dept., DSCE, Bangalore

³UG B.Tech. (CSE) Student of Third Semester, Dept. of Computer Science & Engg., PES

University, Bangalore

⁴Associate Professor, ECE Dept., DSCE, Bangalore, Karnataka

⁵Assistant Professor, ECE Dept., DSCE, Bangalore, Karnataka

⁶Professor & HOD, ECE Dept., DSCE, Bangalore, Karnataka

Abstract

In this paper, we present the implementation of a tic-tac-toe game using the python environment for any gaming application problem. Two players compete in the game of Tic-Tac-Toe, which is played on a 3 by 3 grid. Each participant is given a unique symbol (X or O) to represent the slot they are responsible for filling. The player who first covers a horizontal, vertical, or diagonal row of the board with solely their symbols wins the game. This work, which was written in Python using Pygame on a Jupyter Notebook, suggests a winning Tic-Tac-Toe strategy. This algorithm is made to act like a player while the computer maximises the odds of success by acting in accordance with the model's intelligence. The purpose of this work is to develop a Tic-Tac-Toe strategy in which the game will continue until one player wins and will conclude in a draw if no player selected the optimal spots that may have increased their chances of winning. The work presented here is the mini-project work of the 2nd sem students of electronics & communication engineering department of dayananda sagar college of engg., bangalore.

Keywords—Management, Software, Inventory, App.

1. Introduction

In this section, the design & development of a tic-tac-toe problem solving using python coding is being developed and the results are presented. A brief overview is presented in this introductory note in this context.

2. Role of On-Line Games in Design Process

Before online games were popular, all of us have experienced periods in our lives when offline games like Temple Run, Subway Surfer, and simulation games were the preferred forms of amusement. Even now, playing offline games is still thought to be the best method to pass the time during delays in transportation, including flights, trains, and automobiles. Enter the offline games, and boredom will only be temporarily relieved. Even tech goliaths like Google released its T-rex game in 2014, which was created by only 5 people. That year, it nearly received 20 million plays, which compelled Google to form a separate team specifically for this game. The most popular game in the US in 2012 was an offline game created by an Indian startup called Games2win named "Parking Frenzy." We made the decision to use Python to create a different well-known game that can be played against the computer or in Player vs. Player (PVP) mode [1].

3. Gaming Strategy Developed

The "X" and "O" players in the Tic Tac Toe game alternately mark the squares on a 3 by 3 grid. The game is won by the person who successfully places three of their assigned markings in a row that is

either horizontal, vertical, or diagonal. Whether you're waiting in line or spending time with your kids, playing Tic Tac Toe is a terrific way to pass the time. Save trees by reducing paper consumption. Tic Tac Toe is frequently used as a pedagogical tool to teach the notions of good sportsmanship and the branch of artificial intelligence due to its simplicity [2].

4. Adoption of proposed methodologies

Tic-tac-toe has the following rules given in the form of an algorithm as follows.

All grid positions are unfilled at the beginning of the game.

The players alternately place their figures into open spaces as they turn. Characters "player1" and "player2" (or the AI itself) always place are "X" and "O," respectively.

Characters will never be placed in filled spots by the player.

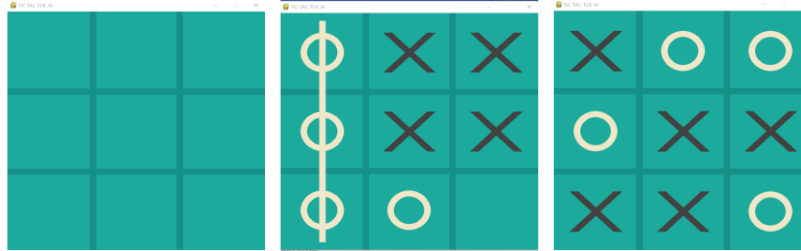


Fig. 1 : Tic Tac Toe patterns

Start Screen for game ; When any of the player win.(Player 1, Player2, or AI) ; In case of a Drawn Match

5. Design of the Coding Process

The code is made up of a "Console board," which resembles a Tic-Tac-Toe board but has zeros in place of the squares. There are basically no uses for this board [3].

6. Design of gaming for employment purpose

The game is designed for continuous enjoyment and passions. Because if the player is not fully aware and notices the saucer fire, he or she must be hit by the saucer-bombs, the game educates the player to be vigilant in any situation they encounter. Despite being an action game, the suggested game does not directly include violence. The game does not involve murdering zombies, animals, or people. As a result, it can also be thought of as a game without violence. Because of the game's straightforward design and basic control scheme, even young children may play it by simply touching a few nearby keyboard keys. The fact that the game is offline adds to its effectiveness as a boredom buster. A long journey, traffic, a flight delay, a tiresome family visit with your mother, and many other things. a number of situations. It might happen at any time or place. Lack of internet access or slow internet exacerbates the problem. present the offline games. You can only escape from such horrible circumstances through offline games. The game excels in this area [4].

7. Development of the Min-Max Algorithm

In this section, the min-max algorithm is used for the development of the program for solving the tic-tac-toe problem. The programme employs the MINI-MAX Algorithm when playing with AI to make the AI's moves. The MINI-MAX algorithm/function, which consists of a collection of functions that keeps playing and exploring future possible states until it reaches a terminal state resulting in a draw, a victory, or a loss, is used in the work given to solve the tic-tac-toe problem. In other words, it lessens the likelihood that a player would prevail when playing against AI. The strategy developed is as follows.

8. Module evaluation

Debug messages are used to ensure that the written code generates the desired outcomes during the coding process. The need that the code compile without any errors is crucial [5].

9. Testing for integration

It is carried out following the completion of module testing to confirm that each module can function properly with the others. When all the fixes are finished, integration testing demonstrates that the system functions as an integrated unit [9].

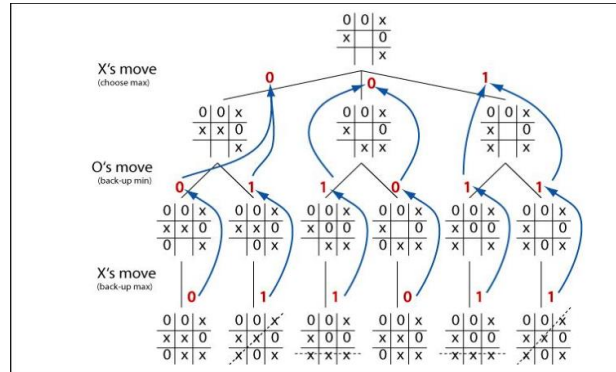


Fig. 1 : Use of min-max function to solve the problem

10. System Evaluation

Here, we evaluate the system for its overall performance and arrive at the conclusive remarks with justifications. The functional testing and usability testing are two of the steps. These will function once the product is in its finished state. The tester will determine whether the product satisfies the game's criteria during the functional test phase. The purpose of the usability test is to determine how simple it is to pick up the game. Any non-team member will take this examination by participating in the game [7].

11. Program Design

Python 3.0 Jupyter Notebook (pip Installed) [1]-[9]

Data scientists can create and share documents that combine live code, equations, computational output, visualisations, and other multimedia elements with explanatory text using the open-source web tool known as the Jupyter Notebook [6].

12. Conclusions & final comments

The following outcomes have been successfully implemented.

Players have the option of performing for initial move.

Players' moves are executed and shown on the game board and console board at the same time.

Player 2 may move and place the symbols assigned to them on the tiles they choose.

The computer chooses its location and shows it while playing with AI.

After each stage, the algorithm continues to look for potential winning circumstances.

If three identical symbols are discovered in quick succession, the game shows which player has won by drawing a line over the matching symbols.

One should not encounter any difficulties because the Python Tic Tac Toe game was constructed utilising only the fundamental Python functionalities. For displaying assistance messages, we have used standard if-else conditions as well as function calls based on user input. All age groups are most accustomed to playing Tic Tac Toe. Any person who makes decisions for a specific reason can possess intelligence. This fundamental notion has been advanced frequently. A tested and described algorithm for playing tic tac toe operates effectively. In general, there are hardly any bugs in the system [8].

References

[1]. D. Pavithra G., Playing Smart – AI, Notion Press, India

[2]. Dr. Pavithra G., et.al., DL and it's techniques, Notion Press, India



- [3]. Dr. Pavithra G., et.al., Computational Intelligence, Notion Press, India
- [4]. Dr. Pavithra G., et.al., ML for Web Applications, Notion Press, India
- [5]. Dr. Pavithra G., et.al., System Software, Mahi Publications, India
- [6]. <https://www.udemy.com/course/complete-python-bootcamp/learn/lecture/9478298#content>
- [7]. <https://www.udemy.com/course/the-art-of-doing-video-game-creation-with-python-and-pygame/>
- [8]. http://www.tutorialspoint.com/python/tk_canvas.htm
- [9]. Python Tutorial: Classes - Odds and Ends