

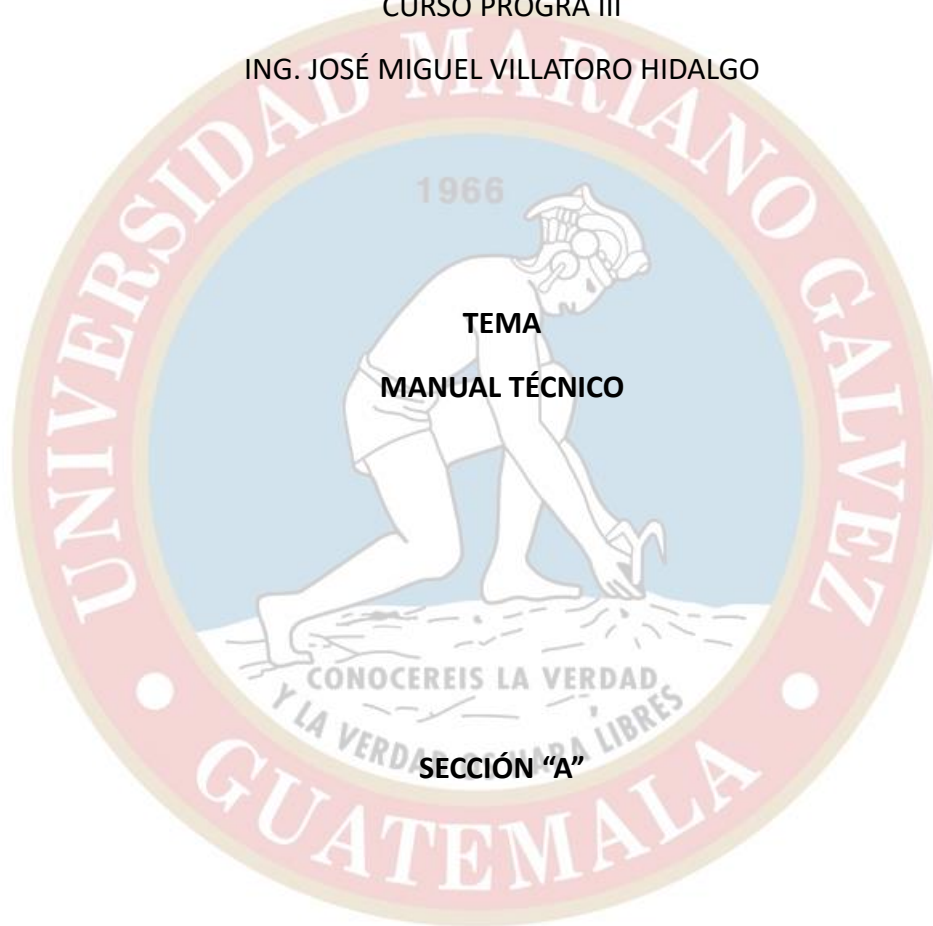
UNIVERSIDAD MARIANO GÁLVEZ DE GUATEMALA

CENTRO UNIVERSITARIO EL NARANJO, MIXCO

FACULTAD DE INGENIERÍA EN SISTEMAS

CURSO PROGRA III

ING. JOSÉ MIGUEL VILLATORO HIDALGO



Nombre:

Carné

Diego Antonio Beteta García

9490-22-12878

Luis Eduardo González Alvarado

9490-22-14408

Keily Andrea Tobar Morales

9490-22-4796

Alan Billy Baten Guigui

9490-22-17906

Manual Técnico

Introducción

Este manual técnico nos da detalles sobre la arquitectura, diseño e implementación del juego del totito con inteligencia artificial. Está destinado a desarrolladores que deseen mejorar el código, para que la IA pueda ir aprendiendo por partidas

Arquitectura del Programa

El sistema está compuesto por las siguientes clases principales:

Board: Representa el tablero del juego y maneja la lógica del juego.

AI: Implementa la inteligencia artificial que juega contra el jugador humano utilizando el algoritmo Minimax.

Clases y Métodos

Clase Board

Atributos:

squares (np.ndarray): Matriz que representa el estado del tablero.

marked_sqs (int): Contador de casillas marcadas.

screen (pygame.Surface): La pantalla de Pygame utilizada para renderizar.

Métodos:

__init__(): Constructor que inicializa el tablero.

final_state(show=False): Determina el estado final del juego.

draw_win_line(start_pos, end_pos): Dibuja la línea de victoria en el tablero.

mark_sqr(row, col, player): Marca una casilla en el tablero con el jugador indicado.

empty_sqr(row, col): Verifica si una casilla está vacía.

get_empty_sqr(): Devuelve una lista de casillas vacías.

isfull(): Verifica si el tablero está lleno.

isempty(): Verifica si el tablero está vacío.

get_state(): Devuelve el estado del tablero.

copy(): Crea una copia del tablero.

Clase AI

Atributos:

player (int): El número del jugador que representa la IA (por defecto 2).

moves (int): Un contador de movimientos realizados por la IA.

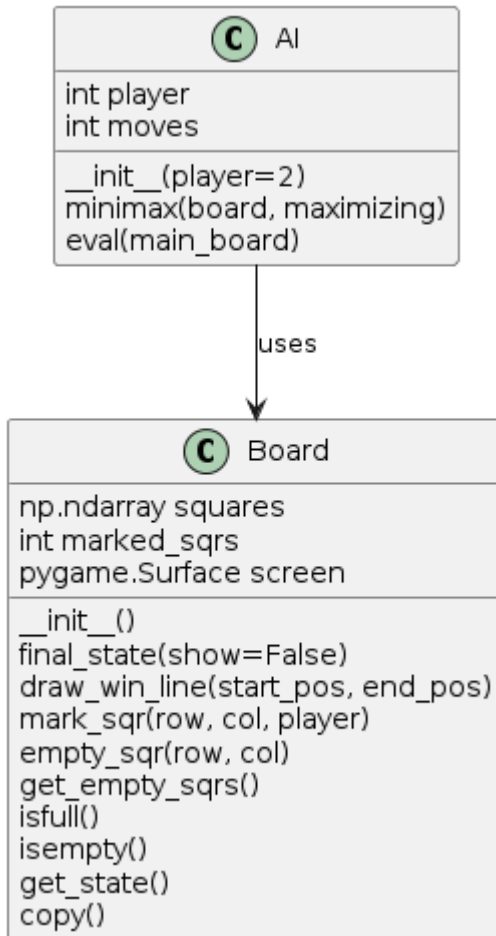
Métodos:

__init__(player=2): Constructor que inicializa la IA.

minimax(board, maximizing): Implementa el algoritmo Minimax para evaluar los movimientos.

eval(main_board): Evalúa el mejor movimiento utilizando el algoritmo Minimax.

Diagrama de Clases: (UML)



Algoritmo Minimax

El algoritmo Minimax es una técnica recursiva utilizada para determinar el movimiento óptimo para la IA. La función minimax evalúa todos los posibles movimientos del tablero y elige el mejor basado en la profundidad de búsqueda y el estado del juego.