**Introdução ao Jogo da Velha com IA**

**Funções Básicas**

No nosso jogo da velha, dois jogadores podem jogar, mas aqui incluímos uma Inteligência Artificial (IA) para controlar um dos jogadores. No caso, o jogador "O" é controlado pela IA, enquanto o jogador "X" é o humano.

Quando você clica em uma célula vazia do tabuleiro, o jogo:

1. Coloca o símbolo do jogador (X ou O) naquela posição.
2. Verifica se alguém venceu.
3. Alterna para o próximo jogador.

**Como a IA Funciona**

A IA, no papel do jogador "O", usa um algoritmo chamado "minimax". Esse algoritmo simula todos os movimentos possíveis e escolhe o melhor para vencer ou empatar o jogo. Isso torna a IA muito "inteligente", pois ela sempre fará o movimento mais estratégico, deixando o jogo mais desafiador.

Por exemplo, se for a vez da IA jogar, ela analisa quais células ainda estão vazias e decide onde colocar o "O" para maximizar suas chances de vencer ou, pelo menos, impedir que o jogador "X" vença.

Vamos explicar o script do jogo da velha com IA em termos de programação:

1. **Definição de elementos HTML**: As variáveis board, cells, e resetButton guardam os elementos do tabuleiro e botão de reset usando getElementById e querySelectorAll.
2. **Estado do Jogo**: Variáveis como currentPlayer, gameActive, e boardState controlam o jogador atual, se o jogo está ativo e o estado do tabuleiro.
3. **Condições de Vitória**: winningConditions é uma matriz que guarda todas as combinações de vitória (linhas, colunas, diagonais).
4. **Manipulação de Cliques**: Quando uma célula é clicada (handleCellClick), o jogo verifica se a célula já foi ocupada ou se o jogo acabou. Se estiver tudo certo, ele atualiza a célula e verifica se há vitória ou empate.
5. **Alternar Jogador**: switchPlayer troca entre os jogadores "X" e "O".
6. **Verificação de Resultado**: A função checkResult compara o estado do tabuleiro com as condições de vitória e, se houver um vencedor, o jogo termina.
7. **Movimento da IA**: Quando é a vez da IA, a função makeAIMove usa o algoritmo **minimax** para decidir a melhor jogada, calculando todos os possíveis resultados do jogo.
8. **Algoritmo Minimax**: Esta função avalia as jogadas disponíveis e simula todos os resultados possíveis, atribuindo pontuações às jogadas e escolhendo a melhor opção para a IA.
9. **Reset**: A função resetGame reinicia o jogo, limpando o tabuleiro e as variáveis.

Isso permite que a IA jogue de forma estratégica, garantindo que ela tome as melhores decisões para vencer ou evitar derrota!

Exercício

1. **O que são e como são usados os elementos HTML no script do jogo da velha?**
2. **Como a variável boardState controla o estado do jogo da velha?**
3. **O que é o algoritmo minimax e como ele ajuda a IA a tomar decisões estratégicas no jogo?**
4. **Qual a função das winningConditions no código?**
5. **Como a função handleCellClick gerencia os cliques dos jogadores?**
6. **O que ocorre quando o jogo é reiniciado usando resetGame?**
7. **Por que a IA usa uma pequena pausa antes de realizar seu movimento?**
8. **Como funciona o processo de alternância de jogadores entre "X" e "O"?.**