```python
import math

def print_board(board):
    for row in board:
        print(" | ".join(row))
        print("-" * 9)

def check_winner(board):

    for i in range(3):
        if board[i][0] != " " and all(board[i][j] == board[i][0] for j in range(3)):
            return board[i][0]
        if board[0][i] != " " and all(board[j][i] == board[0][i] for j in range(3)):
            return board[0][i]

    if board[0][0] != " " and all(board[i][i] == board[0][0] for i in range(3)):
        return board[0][0]
    if board[0][2] != " " and all(board[i][2-i] == board[0][2] for i in range(3)):
        return board[0][2]

    if all(board[i][j] != " " for i in range(3) for j in range(3)):
        return "Tie"
    return None

def minimax(board, depth, is_maximizing):
    winner = check_winner(board)
    if winner == "0":
        return 1
    elif winner == "X":
        return -1
    elif winner == "Tie":
        return 0
```

```python
        if is_maximizing:
            best_score = -math.inf
            for i in range(3):
                for j in range(3):
                    if board[i][j] == " ":
                        board[i][j] = "0"
                        score = minimax(board, depth + 1, False)
                        board[i][j] = " "
                        best_score = max(score, best_score)
            return best_score
        else:
            best_score = math.inf
            for i in range(3):
                for j in range(3):
                    if board[i][j] == " ":
                        board[i][j] = "X"
                        score = minimax(board, depth + 1, True)
                        board[i][j] = " "
                        best_score = min(score, best_score)
            return best_score

def best_move(board):
    move = None
    best_score = -math.inf
    for i in range(3):
        for j in range(3):
            if board[i][j] == " ":
                board[i][j] = "0"
                score = minimax(board, 0, False)
                board[i][j] = " "
                if score > best_score:
                    best_score = score
                    move = (i, j)
    return move
```

```python
def tic_tac_toe_ai():
    board = [[" "]*3 for _ in range(3)]
    print("Tic-Tac-Toe Game: You = X, Computer = O\n")
    print_board(board)

    while True:

        try:
            row = int(input("Enter row (0-2): "))
            col = int(input("Enter column (0-2): "))
        except ValueError:
            print("Invalid input! Enter numbers 0-2.")
            continue

        if 0 <= row <= 2 and 0 <= col <= 2:
            if board[row][col] == " ":
                board[row][col] = "X"
            else:
                print("Cell occupied! Try again.")
                continue
        else:
            print("Invalid position! Enter 0-2.")
            continue

        print_board(board)
        result = check_winner(board)
        if result:
            if result == "Tie":
                print("It's a tie!")
            else:
                print(f"{result} wins!")
            break
        move = best_move(board)
```

```python
            if move:
                board[move[0]][move[1]] = "O"
                print("Computer played:")
                print_board(board)
                result = check_winner(board)
                if result:
                    if result == "Tie":
                        print("It's a tie!")
                    else:
                        print(f"{result} wins!")
                    break
tic_tac_toe_ai()
```

```
>>> %Run -c $EDITOR_CONTENT
  Tic-Tac-Toe Game: You = X, Computer = O

   |   |
  ---------
   |   |
  ---------
   |   |
  ---------
  Enter row (0-2): 1
  Enter column (0-2): 1
   |   |
  ---------
   | X |
  ---------
   |   |
  ---------
  Computer played:
  O |   |
  ---------
   | X |
  ---------
   |   |
  ---------
  Enter row (0-2): 2
  Enter column (0-2): 2
  O |   |
  ---------
   | X |
  ---------
   |   | X
  ---------
  Computer played:
  O |   | O
  ---------
   | X |
  ---------
   |   | X
  ---------
```

```
Enter row (0-2): 0
Enter column (0-2): 1
O | X | O
---------
  | X |
---------
  |   | X
---------
Computer played:
O | X | O
---------
  | X |
---------
  | O | X
---------
Enter row (0-2): 2
Enter column (0-2): 0
O | X | O
---------
  | X |
---------
X | O | X
---------
Computer played:
O | X | O
---------
O | X |
---------
X | O | X
---------
Enter row (0-2): 1
Enter column (0-2): 2
O | X | O
---------
O | X | X
---------
X | O | X
---------
It's a tie!

>>>
```