



Universidad de Guadalajara

CUCEI

Seminario de Solución de Problemas de programación

Clave: I5883

Sección: D27

NRC: 42771

2024 A

Resumen del Funcionamiento del Código: Juego de Gato

Mtro. LUIS ANGEL MORALES LOPEZ

Alumno: Jesús Alejandro Montes Águila

Código: 224006131

Ingeniería en Comunicaciones y Electrónica

Fecha: 30 de noviembre de 2024x

1. Introducción

Este programa es una implementación en Java del clásico juego de 'Gato' (Tic-Tac-Toe) para dos jugadores. A continuación, se describe su funcionamiento.

2. Inicialización

El tablero se representa como una matriz de cadenas (`String[][]`) de 3x3 con valores iniciales que indican las posiciones disponibles (e.g., '11', '12').

Se establece un contador de movimientos (movimientos) para controlar el número máximo permitido (9 en total).

Una variable booleana (`turnoJugadorX`) determina de quién es el turno: `true` para el jugador X y `false` para el jugador O.

3. Ciclo Principal

El juego funciona dentro de un bucle `while` que continúa mientras no haya un ganador y no se hayan realizado más de 9 movimientos. En cada iteración:

1. Se muestra el tablero actual mediante el método `mostrarTablero`.
2. Se solicita al jugador actual que introduzca la posición en la que quiere colocar su marca.
3. Se llama al método `marcarTablero` para colocar la marca (X o O) en la posición seleccionada:

- Si la posición es válida y está disponible, se marca en el tablero y se actualizan los turnos y el contador de movimientos.

- Si no es válida (ya ocupada o inexistente), se notifica al jugador y se solicita un nuevo intento.

4. Después de cada jugada válida, se verifica si hay un ganador usando el método `verificarGanador`:

- Si se detecta un ganador, el juego finaliza mostrando el resultado.

Si no hay un ganador después de 9 movimientos, el juego termina en empate.

4. Verificación del Ganador

El método `verificarGanador` comprueba tres condiciones principales:

1. Filas: Si los tres valores en una fila son iguales.
2. Columnas: Si los tres valores en una columna son iguales.
3. Diagonales: Si los valores de cualquiera de las dos diagonales son iguales.

Si encuentra un ganador, devuelve la marca del jugador ganador (X u O); de lo contrario, retorna `null`.

5. Métodos Auxiliares

1. `mostrarTablero`: Imprime el tablero actual en consola, mostrando las posiciones ocupadas por X, O o las disponibles.
2. `marcarTablero`: Cambia el valor en el tablero si la posición seleccionada está disponible. Devuelve `true` si la jugada fue válida y `false` en caso contrario.

6. Resultados Posibles

1. Victoria: Si un jugador logra alinear tres marcas consecutivas (en fila, columna o diagonal), el programa declara su victoria y termina.
2. Empate: Si se realizan 9 movimientos sin que haya un ganador, se declara empate.

7. Interacción con el Usuario

El jugador introduce las posiciones deseadas (e.g., '11' para la esquina superior izquierda). El programa valida las entradas, gestiona los turnos y muestra mensajes claros para guiar al jugador.

8. Ejemplo de Flujo

1. Tablero inicial:
11 12 13
21 22 23
31 32 33
2. Turno de jugador X: elige '11'. El tablero se actualiza:
X 12 13
21 22 23
31 32 33
3. Turno de jugador O: elige '22'. El tablero se actualiza:
X 12 13
21 0 23
31 32 33
4. Continúa hasta que haya un ganador o empate.