Taller de Juego de Tic Tac Toe

Juan Sebastián Jiménez Giraldo
Universidad Sergio Arboleda
Bogotá, Colombia
juan.jimenez@usa.edu.co

Resumen

En este documento se presenta el desarrollo del clásico juego *Tic Tac Toe* (Tres en Raya), creado como parte de un taller de programación en C++. El objetivo fue construir un programa funcional que permita a dos jugadores enfrentarse por turnos, verificar quién gana o si hay empate, y mostrar los resultados de forma clara. Durante el proceso se reforzaron habilidades como la lógica, el manejo de estructuras de datos y la organización del código. También se buscó escribir un programa legible y bien comentado que refleje una forma práctica y sencilla de aplicar los conceptos básicos de programación.

Keywords

Tic Tac Toe, Tres en Raya, Programación, C++, Lógica Computacional, Algoritmos, Estructuras de Datos

ACM Reference Format:
Juan Sebastián Jiménez Giraldo. 2025. Taller de Juego de Tic Tac Toe. In . ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1. Introducción

El juego *Tic Tac Toe*, más conocido como Tres en Raya, es uno de los ejemplos más sencillos y útiles para aprender a programar. A través de este ejercicio se pueden practicar estructuras de control, condiciones, ciclos y manejo de datos. Además, su desarrollo ayuda a pensar de forma lógica y a entender cómo organizar un programa paso a paso.

El propósito principal de este taller fue implementar el juego en el lenguaje C++, permitiendo que dos jugadores participen y que el sistema detecte automáticamente si hay un ganador o un empate. Más allá del juego en sí, el objetivo era poner en práctica la lógica, la validación de datos y el uso de vectores para almacenar información dentro del programa.

2. Diseño de la Solución

2.1. Estructura de datos

Para representar el tablero se usó una estructura dinámica: `std::vector<std::vector<char>>`, que actúa como una matriz de 3x3. Cada casilla guarda un carácter que representa el movimiento de cada jugador o un espacio vacío. Esto facilita el manejo del tablero y hace que el código sea más claro y flexible.

Los valores posibles en cada casilla son:

- `'X'` → Jugador 1.
- `'O'` → Jugador 2.
- `' '` → Casilla vacía.

Esta estructura se actualiza en cada turno, validando que no se sobreescriban posiciones y garantizando que solo se puedan hacer movimientos válidos.

2.2. Lógica general del juego

El juego sigue una estructura simple pero efectiva:

1. Se inicializa el tablero vacío.
2. Cada jugador elige una posición.
3. El sistema revisa si hay tres símbolos iguales seguidos.
4. Si no hay ganador y el tablero se llena, el juego termina en empate.
5. Los resultados se muestran en pantalla.

Esta lógica hace que el programa sea fácil de entender y funcione correctamente sin importar el orden de las jugadas.

2.3. Código principal en C++

A continuación se muestra el código principal que hace funcionar el juego:

Listing 1: Código base del juego Tic Tac Toe

```
#include <iostream>
#include <vector>
using namespace std;

bool hayGanador(const vector<vector<char>>& t) {
    for (int i = 0; i < 3; i++) {
        if (t[i][0] != ' ' && t[i][0] == t[i][1] && t[i
            ][1] == t[i][2])
            return true;
        if (t[0][i] != ' ' && t[0][i] == t[1][i] && t[1][
            i] == t[2][i])
            return true;
    }
    if (t[0][0] != ' ' && t[0][0] == t[1][1] && t[1][1]
        == t[2][2])
        return true;
    if (t[0][2] != ' ' && t[0][2] == t[1][1] && t[1][1]
        == t[2][0])
        return true;
    return false;
}

int main() {
    vector<vector<char>> tablero(3, vector<char>(3, ' '))
        ;
    char jugador = 'X';
    int fila, columna;
    int movimientos = 0;

    while (true) {
        cout << "Turno del jugador " << jugador << endl;
```


Conference'17, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YYYY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

```
        cout << "Ingrese fila (0-2) y columna (0-2): ";
        cin >> fila >> columna;

        if (fila < 0 || fila > 2 || columna < 0 ||
             columna > 2 || tablero[fila][columna] != ' '
             ) {
            cout << "Movimiento inv lido , intente de
                 nuevo." << endl;
            continue;
        }

        tablero[fila][columna] = jugador;
        movimientos++;

        if (hayGanador(tablero)) {
            cout << "  Jugador   " << jugador << " ha
                 ganado!" << endl;
            break;
        }

        if (movimientos == 9) {
            cout << "Empate." << endl;
            break;
        }

        jugador = (jugador == 'X') ? 'O' : 'X';
    }
    return 0;
}
```

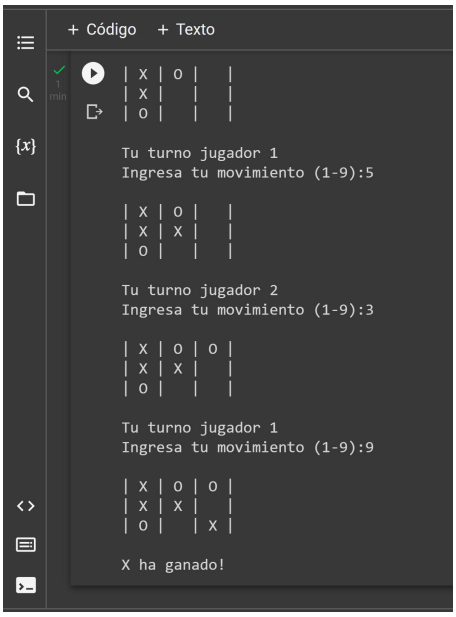

Figura 1: Ejemplo visual del tablero del juego desarrollado en C++.

3. Casos de Prueba

3.1. Victoria por fila

En este caso, el jugador X gana al completar la primera fila del tablero.

- X: [0][0]
- O: [1][1]
- X: [0][1]
- O: [2][1]
- X: [0][2]

El programa detecta correctamente la victoria y muestra el mensaje correspondiente.

3.2. Empate

Cuando ambos jugadores llenan todas las casillas sin formar una línea ganadora, el juego termina en empate. El programa valida correctamente esta condición y muestra el resultado esperado.

4. Conclusiones

Este taller permitió entender cómo aplicar conceptos básicos de programación en un proyecto pequeño pero completo. Trabajar en el desarrollo de *Tic Tac Toe* ayudó a reforzar la lógica de programación, el uso de estructuras dinámicas y la validación de datos dentro de un ciclo.

Además, permitió comprender la importancia de planificar el código antes de escribirlo y de mantener una estructura clara y bien comentada. Aunque es un ejercicio sencillo, demuestra que incluso los programas simples pueden ser una excelente forma de aprender buenas prácticas de desarrollo y resolución de problemas.

Acknowledgments

Agradezco a la Universidad Sergio Arboleda por el acompañamiento y las herramientas brindadas para realizar este proyecto.

Referencias

[1] Bjarne Stroustrup. 2013. *The C++ Programming Language (4th ed.).* Addison-Wesley Professional.
[2] Overleaf. 2024. *ACM Primary Article Template: Guidelines for Authors.* Disponible en: https://www.overleaf.com/latex/templates/acm-primary-article-template/wbvnghbhhzhh
[3] Wikipedia. 2025. *Tic Tac Toe - Historia y reglas.* Disponible en: https://es.wikipedia.org/wiki/Tres_en_l%C3%ADnea

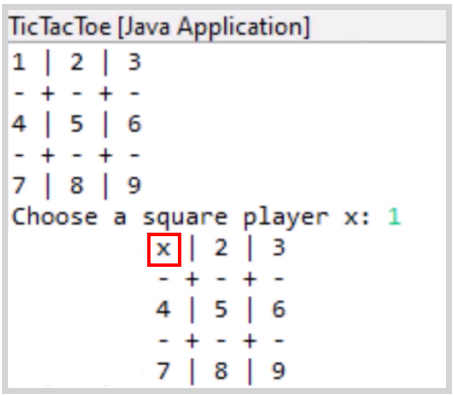

Figura 2: Ejemplo visual del tablero del juego desarrollado en C++.