

Simulador de Torneo de Fútbol

Integrantes del grupo

- Jose Santiago Dominguez Ortiz - jdominguezo@unal.edu.co

- Cristian Camilo Castillo Aguirre - crcastilloa@unal.edu.co

Objetivo general

Desarrollar un simulador en C++ que permita crear un torneo con equipos, fase de grupos y fases eliminatorias. Almacenar los resultados en archivos y generar la tabla de posiciones final, incorporando prácticas de programación estructurada y control de versiones colaborativo en Github.

Objetivos específicos

1. Definir las estructuras de datos (struct) Equipo y Partido, así como los contenedores (vectores y matrices) necesarios para almacenar resultados y clasificaciones.
2. Implementar funciones que generen el calendario de partidos, simulen resultados pseudo-aleatorios y calculen los puntajes de cada equipo.
3. Incorporar al menos una función recursiva para el avance automático de las fases eliminatorias.
4. Manipular punteros para acceder y actualizar dinámicamente los registros de equipos y partidos.
5. Leer y escribir archivos CSV que contengan los equipos, los resultados y la tabla final, garantizando la persistencia de la información.
6. Detectar y gestionar errores de entrada (equipos repetidos, archivos inexistentes, resultados fuera de rango, etc.).
7. Mantener el proyecto en un repositorio remoto utilizando branches, pull requests y merges como parte del flujo de trabajo colaborativo.

Contexto y justificación del problema

Los simuladores deportivos son herramientas valiosas tanto en la industria del entretenimiento como en el análisis de datos. Este proyecto sirve como caso de estudio académico para aplicar conceptos clave del curso de programación: estructuras, manejo de archivos, recursión y buenas prácticas de desarrollo colaborativo. Además, familiariza al equipo con el uso de datasets reales (por ejemplo, plantillas de equipos o históricos de resultados disponibles en Kaggle) y con la disciplina de versionado en Git, competencias demandadas en el ámbito profesional.

Enlace al repositorio remoto

Requisitos funcionales

Descripción
El sistema permitirá registrar N equipos únicos, verificando que no existan nombres duplicados.
El usuario podrá cargar un CSV con la lista de equipos o crearlos manualmente.
Durante la simulación cada partido producirá un resultado aleatorio coherente (0-10 goles por equipo).
Después de cada jornada se actualizará la tabla de posiciones (puntos, GD, GF, GC).
Una vez terminada la fase de grupos, el sistema clasificará a los mejores equipos y ejecutará recursivamente las fases eliminatorias hasta determinar un campeón.
Todos los resultados y la tabla final se almacenarán en archivos CSV.
El programa mostrará mensajes claros sobre el estado de la simulación y advertirá ante entradas inválidas.

Requisitos no funcionales

Descripción
El código debe compilarse con un compilador C++ estándar
El tiempo de simulación para un torneo de hasta 32 equipos no debe exceder los 20 s en un equipo estándar.
La interfaz de usuario será por consola
Se utilizará Git como sistema de control de versiones; las commits deben ser descriptivos.
El repositorio incluirá README.md con instrucciones de compilación y uso, así como ejemplos de entrada y salida.

Historias de usuario

Como...	Quiero...	Para...
Administrador del torneo	Registrar una lista de equipos desde un archivo CSV	Ahorrar tiempo y evitar errores de captura manual.
Administrador del torneo	Iniciar la simulación completa con un solo comando	Obtener rápidamente al campeón y las estadísticas finales.
Aficionado	Visualizar la tabla de posiciones después de cada jornada	seguir la evolución de mi equipo favorito.
Desarrollador	Contar con mensajes de error claros cuando el CSV de equipos está vacío o corrupto	Depurar el sistema con facilidad.
Estadístico deportivo	Exportar los resultados de todos los partidos a un archivo CSV	Poder analizarlos luego en herramientas como Excel