

PROGRAMACIÓN III

Algoritmos y Estructuras de Datos

PARCIAL II

ÍNDICE

ÍNDICE.....	2
Objetivo:.....	3
Modalidad:.....	3
Evaluación:.....	3
Enunciado:.....	4
Requerimientos del cliente:.....	4
1. Eficiencia:.....	4
2. Librerías.....	4
3. Menú.....	4
4. Carga de datos desde archivo CSV:.....	5
5. Procesamiento de datos durante lectura archivo CSV:.....	5
6. Modificación de datos y reprocesamiento:.....	6
7. Consultas dinámicas:.....	6

Objetivo:

- El alumno se desenvuelve profesionalmente en el uso de estructuras y algoritmos de datos mediante la creación de un programa que aborde un desafío cotidiano de la práctica profesional de Ingeniería en Sistema, Informática y Computación y Bioinformática.
- Apropiación de lo aprendido durante el ciclo lectivo de Programación III.

Modalidad:

- Proyecto grupal de la cantidad de miembros anunciada verbalmente en clases.
- Codificación en C++ de un programa. Se pueden usar todos los recursos disponibles: bibliografía, IA, apuntes, etc.
- Codificación se entregará digitalmente en un repositorio de GitHub referenciado en clases antes de la presentación oral.
- **Presentación oral defendiendo el código realizado. Programar defensa oral de 15 minutos siguiendo el siguiente esquema:**
 - Mostrar funcionamiento de las funcionalidades principales (3 minutos).
 - Describir las estructuras y algoritmos principales usados en el código (2 minutos)
 - Mostrar la eficiencia del código y defenderlo oralmente (3 minutos)
 - Preguntas del docente (7 minutos)
 - Nota: prepararse para mostrar las principales partes del código en presentación con diapositivas o debidamente marcadas en el editor de texto.. No se leerá código en extensión durante la presentación.
- Es obligatorio defender oralmente lo codificado. Se computará como ausente quien no asista a la presentación oral así como también quien no presente el código.
- Fecha de entrega de código y fecha de presentación oral se anunciará verbalmente en clases.

Evaluación:

- Se evaluará que el código funcione correctamente en toda su extensión.
- Se evaluará que el código aborde las funcionalidades solicitadas en el enunciado.
- Se evaluará la eficiencia del código.
- Se evaluarán conceptualmente todos los algoritmos y estructuras de datos independientemente de las que se hayan usado o no en la codificación y estén en el programa de la materia. También se evaluará aquellas estructuras y algoritmos que use el estudiante en su código aunque no haya sido abordado en clases.
- Se evaluará la defensa oral.
- Se evaluará la legibilidad del código .
- Se evaluará los comentarios del código.
- Se evaluará las buenas prácticas de programación.

Enunciado:

Desarrolla un programa en C++ que ordene y procese una **base de datos de partidos de fútbol desde un enfoque de eficiencia**. El programa deberá cargar datos de un archivo CSV (Base de datos) y ofrecer diversas funcionalidades a través de un menú.

Requerimientos del cliente:

1. Eficiencia:

El cliente precisa que el programador conmensure la eficiencia de su código contando la **cantidad de condicionales** y el **tiempo de ejecución**. Solicita que ambos cálculos se **muestren por pantalla** luego de la ejecución de procesos principales indicando el nombre del proceso, el algoritmo usado y la estructura usada.

Ej. Procesamiento de datos resultados principales¹ - Tiempo 1ms - Cantidad de if: 30 - estructuras y/o algoritmos usados.

2. Librerías

El cliente facilita librerías de algoritmos y estructuras de datos con las que viene **trabajando en la empresa “Programación III”**². Las pone a disposición y también **flexibiliza que use cualquier librería de C++ externa**. De usar librerías externas el cliente exige que el programador presente en su defensa oral todos los detalles de la librería como lo ha hecho el cliente con sus propias librerías al programador.

3. Menú

El cliente precisa que el programador implemente un Menú con el cual el usuario navegue en las diferentes funcionalidades y que los resultados de las opciones seleccionadas se **muestren claramente y por pantalla contextualizando** adecuadamente con texto y con datos complementarios de la base de datos.

Ej.

Lo que no hay que hacer -> usuario selecciona la opción a del menú. Por la pantalla se ve un número suelto.

Lo que si hay que hacer -> usuario selecciona la opción a del menú.

Por pantalla:

¹ Resultados Principales se refiere al punto 5 de los Requerimientos del Cliente

² Las librerías facilitadas en la cursada de la materia no garantizan la ausencia de errores. Queda a merced del programador sortear esos errores de existir.

La cantidad de goles promedio por partido del Real Madrid en la Champions League es de 2 goles y de 1 gol en la LALIGA.

4. Carga de datos desde archivo CSV:

La base de datos a ordenar y procesar está en formato CSV (Archivo separado por comas) y contiene filas de partidos y con los siguientes datos: **jornada, nombre del equipo local, nombre del equipo visitante, goles del equipo local, goles del equipo visitante, fecha del partido, y competición.**

5. Procesamiento de datos durante lectura archivo CSV:

Durante la lectura³ del archivo CSV el cliente necesita **que se preparen** los siguientes **resultados principales**:

- i. **Los primeros 5 partidos** con mayor cantidad de goles (sumar goles de ambos equipos) y **de cada competición** que haya en la base de datos. De haber más de 5 por igualdad en cantidad de goles, filtrar por fecha, dejando los más recientes. De haber similitud de fechas, filtrar por criterio programador.
- ii. Los **goles totales a favor y en contra** de **cada equipo por competición**. El usuario pedirá que se muestren por pantalla, **de un equipo por vez y deberá mostrar discriminando por competición.**
- iii. **Promedio de goles a favor y en contra** de **cada equipo por competición**. El usuario pedirá que se muestren por pantalla, **de un equipo por vez y deberá mostrar discriminando por competición.**
- iv. **La cantidad total de derrotas y de triunfos** de **cada equipo por competición**. El usuario pedirá que se muestren por pantalla, **de un equipo por vez y deberá mostrar discriminando por competición.**
- v. **La fecha con más goles y la fecha con menos goles** de **cada equipo por competición**. El usuario pedirá que se muestren por pantalla, **de un equipo por vez y deberá mostrar discriminando por competición.**
- vi. **La competición** con más goles convertidos.
- vii. El equipo con **más goles convertidos y el equipo con menos goles convertidos de todas las competiciones y por competición.**

Nota: la muestra de estos resultados se harán cuando el usuario los solicite a través del menú.

³ Este procesamiento inicial deberá anunciarse por pantalla como lo indica el punto 1: cantidad de condicionales, tiempo de ejecución y estructura/s y/o algoritmo/s utilizados.

6. Modificación de datos y reprocesamiento:

El cliente solicita que el usuario podrá agregar, eliminar y modificar los datos. Por ello, el punto 5 deberá ejecutarse automáticamente cada vez que se haga una modificación⁴.

- a. **Agregar un partido**, con fecha, equipo local, goles local, goles visitante, equipo visitante, competición. Guiar al usuario para que no se olvide ningún dato y también bloqueando que introduzca errores a la base de datos.
- b. **Eliminar un partido completo**. Guiar al usuario en la selección del partido a eliminar facilitando con opciones el alcanzar la eliminación.
- c. **Modificar un partido**. Guiar al usuario en la selección del partido a modificar facilitando opciones para alcanzar la modificación. Podrá modificar cualquier campo.

7. Consultas dinámicas:

- a. **Todos los Resultados** de un equipo y en una competición específica ingresado por el usuario.

Ejemplo:

Usuario ingresa: Milán y Champions League

Por pantalla: se listan todos los resultados de ese equipo

En la Champions League:

- Milán 01-01-2023 0 a 1 perdió con Arsenal.
- Milán 02-01-2023 1 a 0 ganó a Austria Wein
- Milán 03-01-2023 3 a 2 ganó a Porto
-
-

- b. **Resultados** de un equipo **entre dos fechas** ingresadas por el usuario.

Ejemplo:

Usuario ingresa: 01-01-2023 al 01-03-2023 Equipo: Real Madrid.

Por pantalla: se listan todos los resultados de ese equipo durante esas fechas (incluidas)

- Real Madrid 01-01-2023 2 a 0 ganó a Roma en Champions League
- Real Madrid 05-01-2023 0 a 0 empató con PSG en Champions League
- Real Madrid 01-03-2023 3 a 0 ganó a Granada CF en LALIGA

⁴ Aplica lo mismo que la nota al pie N° 3.

-

- c. **Comparación de rendimiento general** (cantidad de goles a favor y en contra) entre dos equipos ingresados por el usuario. Ambos equipos no necesariamente se han enfrentado entre ellos y pueden estar en diferentes competiciones. Discriminar por competición.

Ejemplo: Usuario ingresa: Real Madrid y Milán

Por pantalla:

- Real Madrid Goles a favor: 42 Goles en contra: 35 en Champions League
- Milán Goles a favor: 37 Goles en contra 47 en Champions League
- Real Madrid Goles a favor: 25 Goles en contra: 5 en LALIGA.

- d. **Comparación de rendimiento particular** de dos equipos. Analizando sólo los partidos en los que se han enfrentado esos dos equipos **indicar cantidad de partidos que han jugado en contra, la cantidad de empates y cuál de los dos ha ganado más partidos.**

- e. **Umbral:** el usuario podrá ingresar un umbral de cantidad de **goles convertidos por partido en promedio**, especificar si quiere por arriba o por debajo de ese umbral y el programa deberá filtrar a todos los equipos de todas las competiciones que hayan convertido dentro de ese umbral, incluyéndose. Discriminar por competición

Ejemplo:

Usuario ingresa umbral de 3 como máximo.

Por pantalla: se listan todos los equipos que hayan convertido de 0 a 3 goles en promedio por partido:

- Porto 3 goles promedio por partido en Champions League
- Milán 1 gol promedio por partido en Champions League
- Real Madrid 2 goles promedio por partido en Champions League
- Real Madrid 1 gol promedio por partido en LALIGA
-
-