



**Universidad Autónoma de Nuevo León**  
**Facultad de Ciencias Físico Matemáticas**  
**(FCFM)**

**2do Semestre**

**Programación Básica**

**Docente: Perla Marlene Viera Gonzalez**

**PIA E3**

**Equipo 8**

| <b>Nombre</b>                   | <b>Matricula</b> |
|---------------------------------|------------------|
| Ángel Ernesto García<br>Ramírez | 2049907          |
| Aldo Manuel Gómez<br>Rendon     | 1996008          |

**Aula: 403**

**Grupo: 073**

**A entregar el 13 de Mayo del 2024 en Ciudad Universitaria,  
San Nicolas, Nuevo Leon**

## Documentación.

En el avance de esta semana fue algo rápido debido a que eran continuaciones de lo ya realizado la anterior semana por lo que el tiempo de realización fue corto, a continuación se explicara el avance que realizamos.

- *Lectura y apertura de archivos locales:*

El script inicia su ejecución identificando y accediendo a los archivos generados previamente. Estos archivos contienen datos valiosos como el nombre de los equipos, los resultados de los partidos (victoria, empate, derrota), y el estado de cada encuentro (jugado o pendiente).

El script garantiza que la apertura de estos archivos se realice de manera segura, usando mecanismos que previenen errores comunes como rutas incorrectas, archivos vacíos o datos mal formateados. Esto es esencial para asegurar la integridad del proceso y evitar interrupciones durante la ejecución.

- *Interpretación y transformación de datos:*

Una vez abiertos los archivos, el script toma el contenido y lo transforma desde su forma original (habitualmente texto plano) a estructuras de datos internas más eficientes, como listas, diccionarios u objetos según la lógica del programa.

Esta transformación permite que los datos sean manipulados, filtrados o reorganizados de acuerdo con las necesidades analíticas. Por ejemplo, se pueden agrupar partidos por equipo, calcular estadísticas de victorias y derrotas, o preparar resúmenes de rendimiento.

- *Validación de datos cargados:*

Parte crucial del avance es la verificación de que todos los datos esperados hayan sido correctamente cargados. El script realiza comprobaciones para asegurarse de que no faltan registros, que los formatos de los datos coinciden con lo esperado y que no hay errores de lectura. Esta etapa es fundamental para evitar problemas en las fases posteriores de análisis.

- *Optimización del acceso local:*

Al trabajar sobre archivos locales en lugar de depender constantemente de la API externa, el proyecto mejora notablemente en velocidad y eficiencia. Esto significa que se pueden realizar múltiples pruebas, ajustes y análisis sin incurrir en costos adicionales ni

en riesgos de conectividad o límites de uso de la API. Además, se evita la repetición innecesaria de procesos costosos de descarga, enfocándose solo en la reutilización y análisis.

- *Preparación para análisis avanzados:*

Este segundo script no solo carga los datos, sino que los organiza de manera que queden listos para análisis más profundos:

El avance asegura que los datos estén correctamente alineados para ser integrados en módulos futuros del proyecto, que podrían incluir visualización gráfica, exportación a otros formatos (Excel, CSV, bases de datos), o conexión con interfaces de usuario.

- *Análisis estadístico.*

La problemática a resolver hace que diversas métricas de análisis estadístico como la media/mediana, no nos sean de utilidad, por lo que, para el código únicamente conseguimos la moda de los datos. Para ello, tenemos una lista donde las victorias se registran con un 1, las derrotas con un 2 y los empates con un 3, utilizando el modulo statistics, usamos el método .mode() y pasamos como parámetro esta lista, que nos devolverá el dígito que mas se repite, por lo que de esta manera sabremos la moda de los resultados obtenidos para ambos equipos.

- *Preparación para visualización*

Los datos que necesitaban llevar una preparación para ser visualizados únicamente eran los porcentajes de los resultados de ambos equipos, ya que, es la información que ocupamos para resolver nuestra problemática planteada inicialmente. Para preparar estos datos los guardamos en un array de dos dimensiones para que su manipulación sea mas sencilla mas adelante y aparte para aplicar la librería de Numpy en el código.