



**UNIVERSIDAD MARIANO GÁLVEZ DE GUATEMALA CAMPUS
JUTIAPA, JUTIAPA
FACULTAD DE INGENIERÍA EN SISTEMAS DE INFORMACIÓN
CURSO: PROGRAMACION I
DOCENTE: ING. RULDIN EFRAIN AYALA RAMOS**

Laboratorio 3

Kathya Sofía Melgar Marroquín.

ANALISIS SOBRE EL CODIGO

Al visualizar el código llegamos a la conclusión de que, en el código, hay varias tareas repetitivas y es mucho mejor encapsularlas en funciones. Algunas de estas son:

- Agregar estudiantes
- Mostrar la lista de estudiantes
- Calcular promedio de calificaciones
- Mostrar al estudiante con la calificación más alta

Decisiones sobre las variables locales y las variables Globales

Decidimos colocar como variables globales a las variables estudiante y calificaciones ya que deben de ser accesibles desde diferentes funciones.

Y colocamos las variables nombre, calificación, suma promedio, maxCalificacion y estudianreMax como variables locales porque solo se utilizan dentro de funciones específicas.

3. Respuestas a las Preguntas Guía

1. ¿Qué ventajas tiene dividir el código en funciones?

Mejor organización: El código se vuelve más fácil de entender y gestionar, ya que cada función se encarga de una tarea específica.

Reutilización: Las funciones pueden ser reutilizadas en distintas partes del programa sin necesidad de escribir el mismo código repetidamente.

Facilidad de mantenimiento: Si se necesita modificar una parte del programa, se hace de manera centralizada en la función correspondiente, lo que evita errores en otras secciones del código.

2. ¿Por qué es importante limitar el uso de variables globales?

Reducción de errores: Las variables globales pueden ser modificadas en cualquier parte del programa, lo que aumenta el riesgo de errores y dificultad para depurar el código.

Encapsulamiento: Usar variables locales dentro de funciones hace que el código sea más modular y aislado, lo que mejora la claridad y el control.

3. ¿Cómo se puede mejorar la legibilidad del código?

Nombres descriptivos: Usar nombres claros y descriptivos para funciones y variables facilita la comprensión del código, como `AgregarEstudiante()` o `CalcularPromedio()`.

Comentarios: Agregar comentarios en el código que expliquen qué hace cada parte del programa.

Estructura coherente: Mantener un estilo uniforme, con una correcta indentación y organización del código, mejora su legibilidad.

4. Mejoras Adicionales

Validación de Entradas del Usuario

- Sería útil agregar validaciones para asegurarnos de que las entradas del usuario sean correctas.

Función con efecto máquina de escribir

- Solo para que los títulos se vean más estéticos

Agregación de una nueva variable global

- Llamada `usuarioAutenticado` solo para que nos permita tener un usuario y una contraseña y que este mismo la valide para que nos permita iniciar sesión en el programa.

LINK DEL REPOSITORIO: <https://github.com/KSofy/Laboratorio3.git>