**UNIVERSIDAD MARIANO GÁLVEZ DE GUATEMALA**

**FACULTAD DE INGENIERÍA EN SISTEMAS DE INFORMACIÓN Y CIENCIAS DE LA COMPUTACIÓN**



**Programación I**

**Pedro José Marroquín González**

**Nery Osberto Esquivel Palma**

**Ing. Ruldin Ayala**

**21/02/2025**

1. Análisis del Programa

Se identificaron varias funciones que podían encapsularse en módulos para mejorar la organización y estructura del código. Entre ellas destacan:

* 1. Agregar estudiante
  2. Mostrar lista de estudiantes
  3. Calcular promedios
  4. Encontrar al estudiante con la calificación más alta

1. Variables Locales y Globales

Es fundamental determinar cuándo es apropiado utilizar variables locales y cuándo optar por variables globales para optimizar la eficiencia y el mantenimiento del código.

* 1. ¿Qué datos deben ser accesibles en todo el programa?

Datos compartidos entre múltiples funciones o clases.

Configuraciones globales, como constantes.

En programas pequeños donde la modularidad no es un problema.

* 1. ¿Qué datos solo son necesarios dentro de una función específica?

Datos utilizados exclusivamente dentro de una función.

Variables temporales que no deben afectar otras partes del código.

Para mejorar la legibilidad y evitar modificaciones accidentales.

1. Modularización
   1. Conversión del Programa en Funciones

Se modularizó el código correctamente con las siguientes funciones:

* + 1. agregarEstudiante()
    2. mostrarEstudiantes()
    3. calcularPromedio()
    4. mostrarEstudianteConMaxCalificacion()
  1. Definición de Variables Locales y Globales
     1. ¿Qué datos necesitan ser compartidos entre múltiples funciones?
        1. Nombre de los estudiantes.
        2. Calificaciones.
     2. ¿Qué datos sólo son relevantes dentro de una función específica?
        1. En este caso, todas las variables son necesarias en múltiples funciones, ya que tanto los nombres como las calificaciones son esenciales en todo el programa.

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

Permite un código más organizado y fácil de entender, facilitando su reutilización y mantenimiento a futuro. Además, mejora la eficiencia del programa, haciéndolo más escalable y profesional.

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

El uso excesivo de variables globales puede generar errores difíciles de depurar y dificultar la compilación, ya que cualquier función podría modificarlas accidentalmente.

* 1. ¿Cómo se puede mejorar la legibilidad del código?

Manteniendo un código limpio y estructurado.

Utilizando una nomenclatura adecuada para variables y funciones.

Agregando comentarios para facilitar su comprensión.