**Imagen que contiene Logotipo

El contenido generado por IA puede ser incorrecto.**

**UNIVERSIDAD MARIANO GALVEZ, CAMPUS JUTIAPA**

**CURSO: PROGRAMACIÓN I**

**CATEDRÁTICO: ING. RULDIN AYALA.**

**ARANZA BRIGITTE RUEDA ALVARADO**

No. De carné: 0905-24-7854

**21 de febrero de 2025**

**Análisis del programa Sistema de gestión de estudiantes:**

1. Al ingresar estudiantes nos pide únicamente un nombre y una calificación y no nos especifica más.
2. No tienen un límite de nota de calificación, aceptando cualquier número, seria recomendable establecer un limite y si se ingresar un número mayor a 100 debería emitir un mensaje que diga que es un numero invalido y regrese al menú principal, observando que todo se almacena en un alista de tipo string.
3. Falta de datos importantes como el nombre completo y las notas de semestre, sección y carrera .
4. En el menú calcular promedio de calificaciones se debería usar una variable local para cada estudiante.
5. Todos los datos de las calificaciones las está almacenando en una sola variable y el promedio de las calificaciones.
6. El static lista crea una nueva lista, lo cual guarda los archivos de forma temporal y se borran cuando dejamos de ejecutar el programa.

Cuando es mejor usar una variable local? Cuando se debe ser más específico en lo que queramos y se necesiten datos únicos.

Cuando es mejor usar una variable global? Cuando queramos almacenar datos que no sean de suma importancia.

* ¿Qué ventajas tiene dividir el código en funciones?

Dividir el condigo en funciones nos suele ser útil para llevar el código de una manera más organizada y también para simplificar nuestro código haciendo uso de la llamada a la función.

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

Es importante porque las variables globales pueden ser sobrescritas de forma accidental, causando conflictos. Además se toma en cuenta que el código con muchas variables globales es más difícil de mantener y entender.

* ¿Cómo se puede mejorar la legibilidad del código?

Usar nombres de variables y funciones que describan su propósito, dividir el código en funciones pequeñas y especificas y tener un estilo de codificación consistente en cuanto a nombres y formatos.