

UNIVERSIDAD MARIANO GALVEZ DE GUATEMALA
INGENIERIA EN SISTEMAS DE INFORMACION Y CIENCIAS DE LA
COMPUTACION CAMPUS JUTIAPA



TECER SEMESTRE

SECCION: "A"

- Robertson Alessandro Muñoz Paredes
Carné: 0905-24-16433
- Mariana Alejandra García Hernández
Carné: 0905-24-24315

PROGRA1

EJERCICIO CLASE #3

1. ANALISIS DEL PROGRAMA

Tareas repetitivas que pueden encapsularse en funciones

- **Identificar las tareas repetitivas o que pueden encapsularse en funciones. Por ejemplo:**
 - **Agregar estudiantes.**
 - **R//** acá le pide al usuario que ingrese su nombre y su calificación, podemos agregar varios estudiantes creo que no tiene ningún limite
 - **Mostrar la lista de estudiantes.**
 - **R//** la lista de estudiantes es la segunda opción acá despliega a cuantos estudiantes agregamos, aparece su nombre y la nota que el usuario coloco
 - **Calcular promedios.**
 - **R//** tercera función acá suma todas las calificaciones y luego las divide por los cursos que tenemos
 - **Encontrar al estudiante con la calificación más alta.**
 - **R//** acá supongamos agg 2 estudiantes Alessandro con calificación de 85 y Mariana con calificación de 86 nos despliega la lista y nos indica que Mariana tiene una calificación mas alta

Variables locales vs variables globales

- **Discutir cuándo es apropiado usar variables locales y cuándo usar variables globales.**
 - **Alessandro**

R// yo pienso que ambas variables son demasiados útiles solo que cada una tiene su pro y su contra ya que la variable global uno la declara desde el inicio de la estructura del código y se puede acceder a cada comento con la variable global, en cambio la variable local solo se puede usar y acceder en la función que se declaran
 - **Mariana//**

R// para mi es bueno utilizar la variable global cuando sabemos que será una operación que utilizaremos varias veces, entonces justo ahí podemos aplicar la variable global para optimizar nuestro código. En cambio, la variable local es apropiado usarla para operaciones específicas.

- **¿Qué datos deben ser accesibles en todo el programa?**
- **R//** los datos que deben de ser mas accesibles y a los cuales se le tiene que dar mas prioridad en todo el programa es a lista de estudiantes, ya que si no se tiene ningún estudiante agg no se podrá sacar ni promedios ni estudiante con calificaciones altas etc. y a este dato lo tenemos de declarar como una variable global
- **¿Qué datos solo son necesarios dentro de una función específica?**
- **R//** el dato que es necesario dentro de todas las funciones del código es la del Promedio ya que solo es necesario a la hora de calcular un promedio o de sumar las calificaciones

2. MODULARIZACION

Definir variables locales y globales

- **R//** Las variables globales resultan visibles y disponibles para todas las sentencias de un script. Las variables locales sólo resultan visibles y disponibles dentro de la función en la que están definidas.
- **Defina qué variables deben ser locales y cuáles deben ser globales.**
 - **R//** estas son las variables que son globales
 - `List<string> estudiantes = new List<string>();`
 - `List<double> calificaciones = new List<double>();`
 - **R//** y estas son las variables locales
 - `double maxCalificacion`
 - `estudianteMax`
 - **¿Qué datos necesitan ser compartidos entre múltiples funciones?**
 - **R//** los que usaremos más frecuentes
 - **¿Qué datos solo son relevantes dentro de una función específica?**
 - **R//** los que asignamos o serán utilizados 1 s

3. PREGUNTAS GUIAS

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

- **R//** se organiza mejor la estructura del código y se trabaja de una forma mas clara y limpia, y las funciones las podemos utilizar en varios códigos, se reutiliza

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

- **R//** por la dificultad de legibilidad del código ya que en cualquier parte de programa podemos cambiar el valor de las

variables y eso nos causara errores los cuales tendremos muchas dificultades con las variables

3. ¿Cómo se puede mejorar la legibilidad de un código?

- **R//** nombrando bien la estructura y variables del código y agregando comentarios que nos indiquen lo que ara tal función, y tener una identificación adecuada para las selecciones de variables del programa

MEJORAS ADICIONALES

- **Validación de Entradas al Usuario:**

1. Se modifica la parte donde el usuario ingresa su calificación para asegurarnos de que sea un valor numérico y esté dentro de un rango válido podemos hacer una función (por ejemplo, entre 0 y 100):

Se podría coloca una función como esta

```
Console.Write("Ingrese la calificación del estudiante (0 - 100): "); double
calificacion; while (!double.TryParse(Console.ReadLine(), out calificacion) ||
calificacion < 0 || calificacion > 100) {
```