



UNIVERSIDAD MARIANO GÁLVEZ DE  
GUATEMALA CAMPUS

JUTIAPA, JUTIAPA

FACULTAD DE INGENIERÍA EN SISTEMAS  
DE INFORMACIÓN

CURSO: Emprendedores de Negocios

DOCENTE: Ingeniero Ruldin Efraín Ayala  
Ramos

Alumno: Jostyn Manrique Godoy  
Chinchilla



- El Inge y yo Viendo porque no compila mi código  
(Me faltaba una coma)

## Conceptos principales del código:

### 1. Main (Función Principal)

- Es el punto de entrada del programa
- Contiene un menú interactivo que se ejecuta en bucle (while)
- Maneja la selección de opciones del usuario

### 2. Calculadora

- Permite realizar operaciones básicas (suma, resta, multiplicación, división)
- Maneja dos números y devuelve un resultado
- Incluye validación para división por cero

### 3. ValidarContraseña

- Comprueba si la contraseña ingresada coincide con "1234"
- Usa un bucle do-while para pedir la contraseña hasta que sea correcta

### 4. NumerosPrimos

- Verifica si un número es primo (divisible solo por 1 y por sí mismo)
- Utiliza un algoritmo de verificación optimizado

### 5. ContarVocales

- Cuenta cuántas vocales hay en un texto
- Convierte el texto a minúsculas para facilitar el conteo

## Palabras clave importantes:

### 1. try-catch

- *Significado:* Estructura para manejar errores
- *Ejemplo en el código:* Se usa para evitar que el programa se cierre si el usuario ingresa datos incorrectos

### 2. switch

- *Significado:* Estructura de control que permite múltiples caminos según un valor
- *Uso:* Maneja las diferentes opciones del menú

### 3. **void**

- *Significado:* Indica que una función no devuelve ningún valor
- *Ejemplo:* static void Calculadora()

### 4. **parse**

- *Significado:* Convertir un tipo de dato a otro
- *Ejemplo:* int.Parse() convierte texto a números

### 5. **Random**

- *Significado:* Clase para generar números aleatorios
- *Uso:* En el juego de adivinanza para crear el número secreto

### 6. **foreach**

- *Significado:* Bucle que recorre cada elemento de una colección
- *Uso:* Se usa en ContarVocales para revisar cada letra

### 7. **ToLower()**

- *Significado:* Convierte texto a minúsculas
- *Uso:* Para hacer que el conteo de vocales no sea sensible a mayúsculas/minúsculas

### 8. **break**

- *Significado:* Rompe o sale de un bucle o switch
- *Uso:* Para terminar cada caso del switch

### 9. **continue**

- *Significado:* Salta a la siguiente iteración de un bucle
- *Uso:* Cuando hay un error y se quiere volver al inicio del menú

### 10. **default**

- *Significado:* Caso por defecto en un switch cuando no se cumple ninguna otra opción
- *Uso:* Para manejar opciones inválidas en el menú

```
using System;
```

```
class Program
```

```
{
```

```
    static void Main()
```

```
    {
```

```
        while (true)
```

```
        {
```

```
            try
```

```
            {
```

```
                MostrarMenu();
```

```
                Console.Write("Elija una opción: ");
```

```
                string input = Console.ReadLine();
```

```
                // Validar si la entrada es un número
```

```
                if (!int.TryParse(input, out int opcion))
```

```
                {
```

```
                    Console.WriteLine("\nError: Por favor ingrese un número válido.");
```

```
                    Console.WriteLine("Presione cualquier tecla para continuar...");
```

```
                    Console.ReadKey();
```

```
                    continue;
```

```
                }
```

```
                // Validar si el número está en el rango válido
```

```
                if (opcion < 1 || opcion > 11)
```

```
                {
```

```
Console.WriteLine("\nError: Por favor seleccione una opción entre 1 y 11.");  
Console.WriteLine("Presione cualquier tecla para continuar...");  
Console.ReadKey();  
continue;  
}
```

```
switch (opcion)  
{  
    case 1: Calculadora(); break;  
    case 2: ValidarContraseña(); break;  
    case 3: NumerosPrimos(); break;  
    case 4: SumaPares(); break;  
    case 5: ConversionTemperatura(); break;  
    case 6: ContarVocales(); break;  
    case 7: Factorial(); break;  
    case 8: JuegoAdivinanza(); break;  
    case 9: Intercambio(); break;  
    case 10: TablaMultiplicar(); break;  
    case 11:  
        Console.WriteLine("¿Está seguro que desea salir? (S/N)");  
        if (Console.ReadLine().ToUpper() == "S")  
        {  
            Console.WriteLine("¡Gracias por usar el programa!");  
            return;  
        }  
        break;  
}
```

```
}
```

```
    Console.WriteLine("\nPresione cualquier tecla para volver al menú principal...");
```

```
    Console.ReadKey();
```

```
}
```

```
catch (Exception ex)
```

```
{
```

```
    Console.WriteLine($"Error inesperado: {ex.Message}");
```

```
    Console.WriteLine("Presione cualquier tecla para continuar...");
```

```
    Console.ReadKey();
```

```
}
```

```
}
```

```
}
```

```
static void MostrarMenu()
```

```
{
```

```
    Console.Clear(); // Limpia la pantalla para mejor visualización
```

```
    Console.WriteLine("=== Menú de opciones ===");
```

```
    Console.WriteLine("1) Calculadora básica");
```

```
    Console.WriteLine("2) Validación de contraseña");
```

```
    Console.WriteLine("3) Verificar número primo");
```

```
    Console.WriteLine("4) Sumar números pares");
```

```
    Console.WriteLine("5) Conversión de temperatura");
```

```
    Console.WriteLine("6) Contador de vocales");
```

```
    Console.WriteLine("7) Cálculo de factorial");
```

```
Console.WriteLine("8) Juego de adivinanza");
Console.WriteLine("9) Intercambio de valores");
Console.WriteLine("10) Tabla de multiplicar");
Console.WriteLine("11) Salir");
Console.WriteLine("=====");
}
```

```
static void Calculadora()
```

```
{
    try
    {
        Console.Write("Ingrese el primer número: ");
        if (!double.TryParse(Console.ReadLine(), out double num1))
        {
            Console.WriteLine("Error: El primer número no es válido.");
            return;
        }
    }
```

```
    Console.Write("Ingrese el segundo número: ");
    if (!double.TryParse(Console.ReadLine(), out double num2))
    {
        Console.WriteLine("Error: El segundo número no es válido.");
        return;
    }
```

```
    Console.WriteLine("\nOperaciones disponibles:");
```

```
Console.WriteLine("1) Suma");
Console.WriteLine("2) Resta");
Console.WriteLine("3) Multiplicación");
Console.WriteLine("4) División");
Console.Write("\nElija una operación: ");
```

```
if (!int.TryParse(Console.ReadLine(), out int opcion) || opcion < 1 || opcion > 4)
{
    Console.WriteLine("Error: Operación no válida.");
    return;
}
```

```
switch (opcion)
{
    case 1:
        Console.WriteLine($"Resultado: {num1 + num2}");
        break;
    case 2:
        Console.WriteLine($"Resultado: {num1 - num2}");
        break;
    case 3:
        Console.WriteLine($"Resultado: {num1 * num2}");
        break;
    case 4:
        if (num2 == 0)
        {
```



```
        Console.WriteLine("Error: No se puede dividir por cero.");  
        return;  
    }  
    Console.WriteLine($"Resultado: {num1 / num2}");  
    break;  
}  
}  
catch (Exception ex)  
{  
    Console.WriteLine($"Error en la calculadora: {ex.Message}");  
}  
}  
}
```