



UNIVERSIDAD NACIONAL DE EDUCACIÓN  
Enrique Guzmán y Valle – La Cantuta  
*Alma Máter del Magisterio Nacional*

FACULTAD DE CIENCIAS  
Departamento Académico de Matemática e Informática

# 10 PROGRAMAS EN C# (Windows Forms)

---

<b>Curso:</b>	Lenguaje de Programación II
<b>Especialidad:</b>	Matemática e Informática
<b>Sección:</b>	C1
<b>Promoción:</b>	2024-I
<b>Año académico:</b>	2025

---

**Docente:** Mg. Abel Orizano Quedo

**Integrantes:**

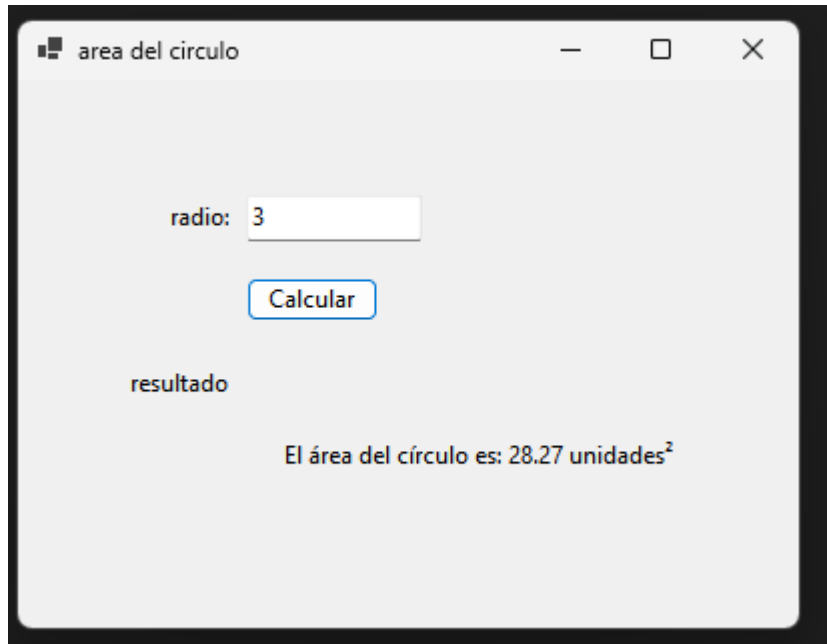
JOHAN JEANPIER SORIANO LUIS,  
WILLIAM ANDERSON ROJAS ROJAS,  
JOSE MARIA REYES RIVERA,  
JUAN JOSE SEBASTIAN DE LA CRUZ PACHECO

## Contenido

Programa 01: Cálculo del área de un círculo .....	3
Código utilizado .....	3
Programa 02: Conversor de temperatura Celsius a Fahrenheit .....	5
Código utilizado .....	5
Programa 03: Comparador de dos números .....	7
Código utilizado .....	7
Programa 04: Cálculo de sueldo final con bonificación .....	9
Código utilizado .....	9
Programa 05: Cálculo de edad según año de nacimiento .....	11
Código utilizado .....	11
Programa 06: Organizador de tareas del día .....	13
Código utilizado .....	13
Programa 07: Calculadora de descuento en tienda .....	16
Código utilizado .....	16
Programa 08: Generador de destino aleatorio .....	19
Código utilizado .....	19
Programa 09: Test de arquetipos Soldado - Poeta - Rey .....	22
Código utilizado .....	22
Programa 10: Simulador de vida diaria .....	25
Código utilizado .....	25

## Programa 01: Cálculo del área de un círculo

Aplicación Windows Forms que calcula el área de un círculo a partir del radio ingresado por el usuario.



### Código utilizado

```
using System;
using System.Windows.Forms;

namespace ejercicio1;

internal static class Program
{
    /// <summary>
    /// Punto de entrada principal para la aplicación.
    /// </summary>
    [STAThread]
    static void Main()
    {
        ApplicationConfiguration.Initialize();
        Application.Run(new Form1());
    }
}

Archivo del formulario (código de eventos):
using System;
using System.Windows.Forms;

namespace ejercicio1
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {

```

```

        // Puedes dejar esto vacío o agregar código de inicialización si lo
necesitas.
    }

    private void label1_Click(object sender, EventArgs e)
    {
        // Evento generado por el diseñador (no es obligatorio usarlo).
    }

    private void label1_Click_1(object sender, EventArgs e)
    {
        // Evento generado por el diseñador (no es obligatorio usarlo).
    }

    private void BtnCalcular_Click(object sender, EventArgs e)
    {
        string textoRadio = txtRadio.Text.Trim();

        // Validar que no esté vacío
        if (string.IsNullOrEmpty(textoRadio))
        {
            MessageBox.Show("Por favor ingrese el radio.", "Dato faltante",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            txtRadio.Focus();
            return;
        }

        // Validar que sea numérico
        if (!double.TryParse(textoRadio, out double radio))
        {
            MessageBox.Show("Ingresa un valor numérico válido para el radio.",
                "Dato inválido", MessageBoxButtons.OK, MessageBoxIcon.Warning);
            txtRadio.Focus();
            return;
        }

        // Validar que sea positivo
        if (radio <= 0)
        {
            MessageBox.Show("El radio debe ser mayor que cero.",
                "Dato inválido", MessageBoxButtons.OK, MessageBoxIcon.Warning);
            txtRadio.Focus();
            return;
        }

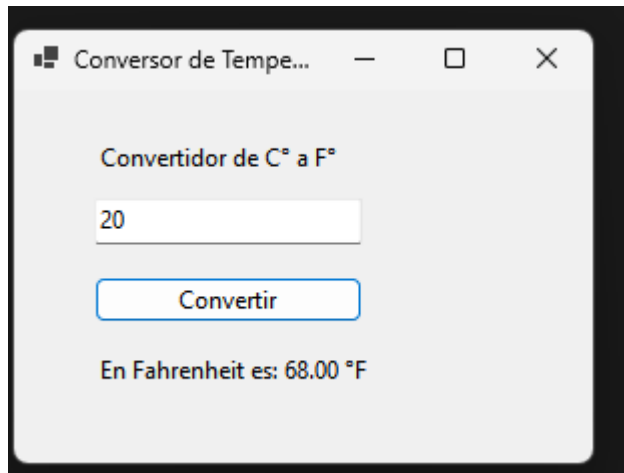
        // Cálculo del área:  $A = \pi * r^2$ 
        double area = Math.PI * Math.Pow(radio, 2);

        lblResultado.Text = $"El área del círculo es: {area:0.00} unidades²";
    }
}

```

## Programa 02: Conversor de temperatura Celsius a Fahrenheit

Aplicación Windows Forms que convierte una temperatura en grados Celsius a Fahrenheit.



### Código utilizado

```
using System;
using System.Windows.Forms;
```

```
namespace TempConverterApp
{
    internal static class Program
    {
        [STAThread]
        static void Main()
        {
            Application.SetHighDpiMode(HighDpiMode.SystemAware);
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}
```

Archivo del formulario (código de eventos):

```
using System;
using System.Windows.Forms;
```

```
namespace TempConverterApp
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        // Botón "Convertir"
        private void button1_Click(object sender, EventArgs e)
        {
            double celsius;

            if (!double.TryParse(textBox1.Text, out celsius))
            {
                MessageBox.Show("Ingrese una temperatura válida en grados Celsius.",
```

```

                                "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error);
        return;
    }

    double fahrenheit = (celsius * 9 / 5) + 32;

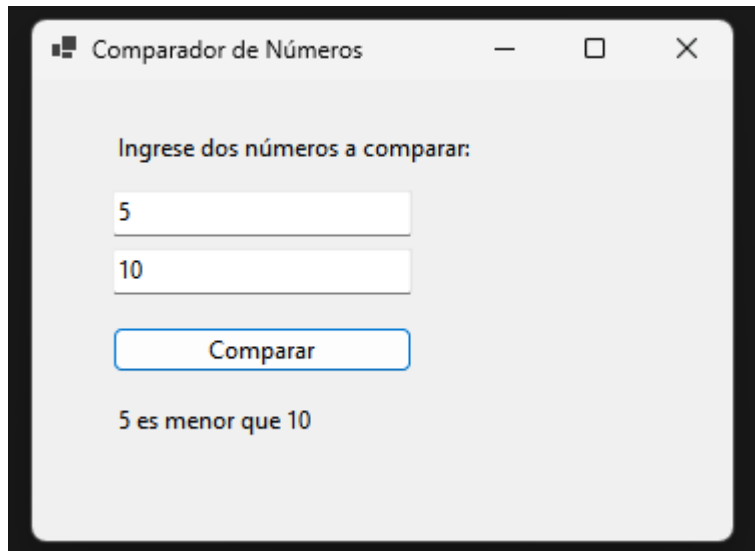
    label11.Text = "En Fahrenheit es: " + fahrenheit.ToString("0.00") + "
°F";
}

private void labelTitulo_Click(object sender, EventArgs e)
{
}
}
}

```

## Programa 03: Comparador de dos números

Aplicación Windows Forms que compara dos números y muestra cuál es mayor, menor o si son iguales.



### Código utilizado

```
using System;
using System.Windows.Forms;

namespace ComparadorNumerosApp
{
    internal static class Program
    {
        [STAThread]
        static void Main()
        {
            Application.SetHighDpiMode(HighDpiMode.SystemAware);
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}
```

Archivo del formulario (código de eventos):

```
using System;
using System.Windows.Forms;

namespace ComparadorNumerosApp
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            // Botón "Comparar"
            private void button1_Click(object sender, EventArgs e)
            {
                double num1, num2;

                if (!double.TryParse(textBox1.Text, out num1) ||
                    !double.TryParse(textBox2.Text, out num2))
```

```

        {
            MessageBox.Show("Ingrese valores numéricos válidos.",
                           "Error", MessageBoxButtons.OK,
                           MessageBoxIcon.Error);
            return;
        }

        string resultado;

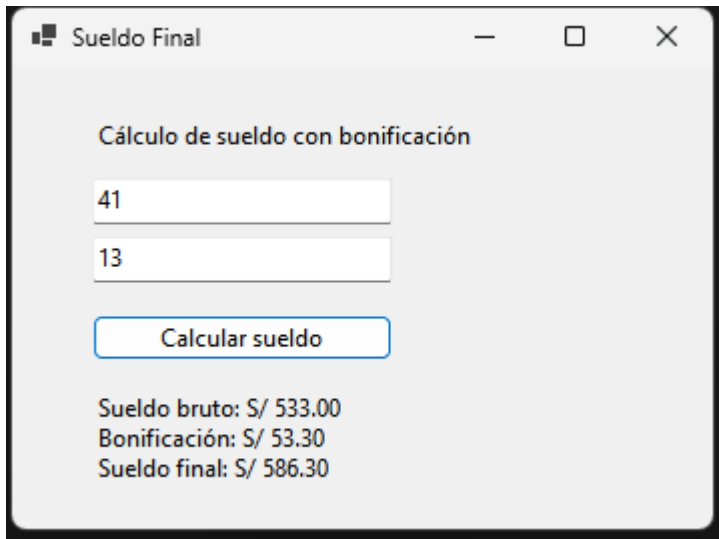
        if (num1 > num2)
            resultado = $"{num1} es mayor que {num2}";
        else if (num1 < num2)
            resultado = $"{num1} es menor que {num2}";
        else
            resultado = "Ambos números son iguales";

        label1.Text = resultado;
    }
}

```

## Programa 04: Cálculo de sueldo final con bonificación

Aplicación Windows Forms que calcula el sueldo bruto y agrega una bonificación del 10% si las horas trabajadas superan 40.



### Código utilizado

```
using System;
using System.Windows.Forms;

namespace SueldoFinalApp
{
    internal static class Program
    {
        [STAThread]
        static void Main()
        {
            Application.SetHighDpiMode(HighDpiMode.SystemAware);
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}
```

Archivo del formulario (código de eventos):

```
using System;
using System.Windows.Forms;

namespace SueldoFinalApp
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        // Botón "Calcular sueldo"
        private void button1_Click(object sender, EventArgs e)
        {
            double horas, tarifa;

            if (!double.TryParse(textBox1.Text, out horas) ||
                !double.TryParse(textBox2.Text, out tarifa))
            {
            }
        }
    }
}
```

```

        MessageBox.Show("Ingrese valores numéricos válidos.",
                        "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error);
        return;
    }

    if (horas < 0 || tarifa < 0)
    {
        MessageBox.Show("Las horas y la tarifa deben ser mayores o iguales
a 0.",
                        "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error);
        return;
    }

    double sueldoBruto = horas * tarifa;
    double bonificacion = 0;

    if (horas > 40)
    {
        bonificacion = sueldoBruto * 0.10; // 10% de bono
    }

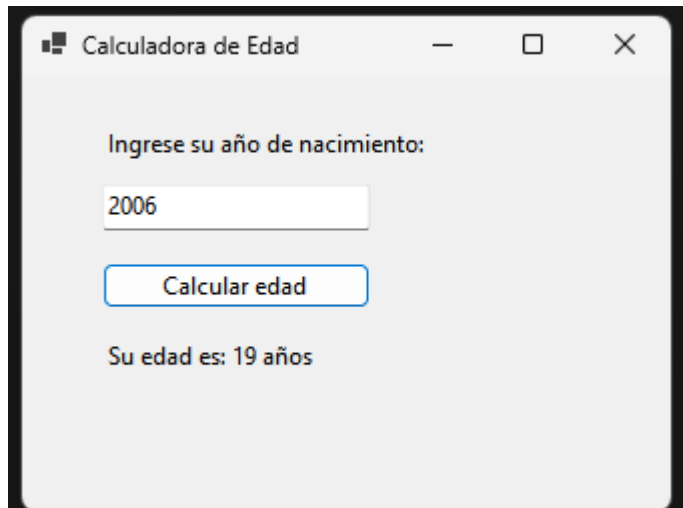
    double sueldoFinal = sueldoBruto + bonificacion;

    label11.Text = $"Sueldo bruto: S/ {sueldoBruto:0.00}
" +
        $"Bonificación: S/ {bonificacion:0.00}
" +
        $"Sueldo final: S/ {sueldoFinal:0.00}";
    }
}

```

## Programa 05: Cálculo de edad según año de nacimiento

Aplicación Windows Forms que calcula la edad aproximada del usuario usando el año de nacimiento.



### Código utilizado

```
using System;
using System.Windows.Forms;

namespace EdadApp
{
    internal static class Program
    {
        [STAThread]
        static void Main()
        {
            Application.SetHighDpiMode(HighDpiMode.SystemAware);
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}
```

Archivo del formulario (código de eventos):

```
using System;
using System.Windows.Forms;

namespace EdadApp
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        // Botón "Calcular edad"
        private void button1_Click(object sender, EventArgs e)
        {
            int nacimiento;

            if (!int.TryParse(textBox1.Text, out nacimiento))
            {
                MessageBox.Show("Ingrese un año válido.",
                                "Error", MessageBoxButtons.OK,
```

```
MessageBoxIcon.Error);
        return;
    }

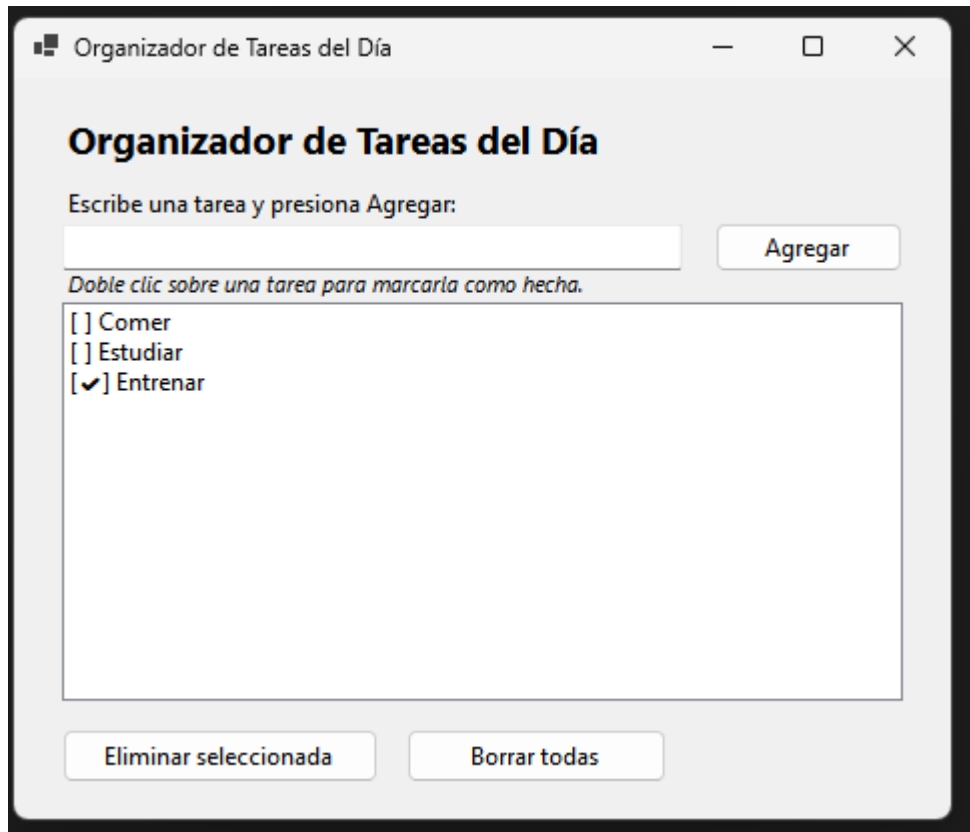
    int edad = DateTime.Now.Year - nacimiento;

    label1.Text = "Su edad es: " + edad + " años";
}

private void Form1_Load(object sender, EventArgs e)
{
}
}
```

## Programa 06: Organizador de tareas del día

Aplicación Windows Forms tipo lista de tareas: permite agregar, eliminar, limpiar y marcar tareas como completadas.



### Código utilizado

```
using System;
using System.Windows.Forms;

namespace OrganizadorTareasDia;

internal static class Program
{
    /// <summary>
    /// Punto de entrada principal para la aplicación.
    /// </summary>
    [STAThread]
    static void Main()
    {
        ApplicationConfiguration.Initialize();
        Application.Run(new MainForm());
    }
}

Archivo del formulario (código de eventos):
using System;
using System.Collections.Generic;
using System.Windows.Forms;

namespace OrganizadorTareasDia
{
    public partial class MainForm : Form
    {
        private readonly List<TareaItem> _tareas = new();
    }
}
```

```

public MainForm()
{
    InitializeComponent();
}

private void btnAgregar_Click(object sender, EventArgs e)
{
    string texto = txtTarea.Text.Trim();

    if (string.IsNullOrEmpty(texto))
    {
        MessageBox.Show("Escribe una tarea antes de agregarla.", "Tarea
vacía",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        txtTarea.Focus();
        return;
    }

    _tareas.Add(new TareaItem { Texto = texto, Completada = false });
    txtTarea.Clear();
    txtTarea.Focus();
    RefrescarLista();
}

private void btnEliminarSeleccionada_Click(object sender, EventArgs e)
{
    int index = lstTareas.SelectedIndex;
    if (index < 0 || index >= _tareas.Count)
    {
        MessageBox.Show("Selecciona una tarea para eliminarla.", "Sin
selección",
            MessageBoxButtons.OK, MessageBoxIcon.Information);
        return;
    }

    _tareas.RemoveAt(index);
    RefrescarLista();
}

private void btnLimpiarLista_Click(object sender, EventArgs e)
{
    if (_tareas.Count == 0)
    {
        return;
    }

    var result = MessageBox.Show("¿Seguro que deseas borrar TODAS las
tareas?",
        "Confirmar", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

    if (result == DialogResult.Yes)
    {
        _tareas.Clear();
        RefrescarLista();
    }
}

private void lstTareas_DoubleClick(object sender, EventArgs e)
{
    int index = lstTareas.SelectedIndex;
    if (index < 0 || index >= _tareas.Count)
    {
        return;
    }
}

```

```

        // Cambiar estado de completada
        _tareas[index].Completada = !_tareas[index].Completada;
        RefrescarLista();
    }

    private void RefrescarLista()
    {
        lstTareas.Items.Clear();
        foreach (var t in _tareas)
        {
            string estado = t.Completada ? "[✓] " : "[ ] ";
            lstTareas.Items.Add(estado + t.Texto);
        }
    }

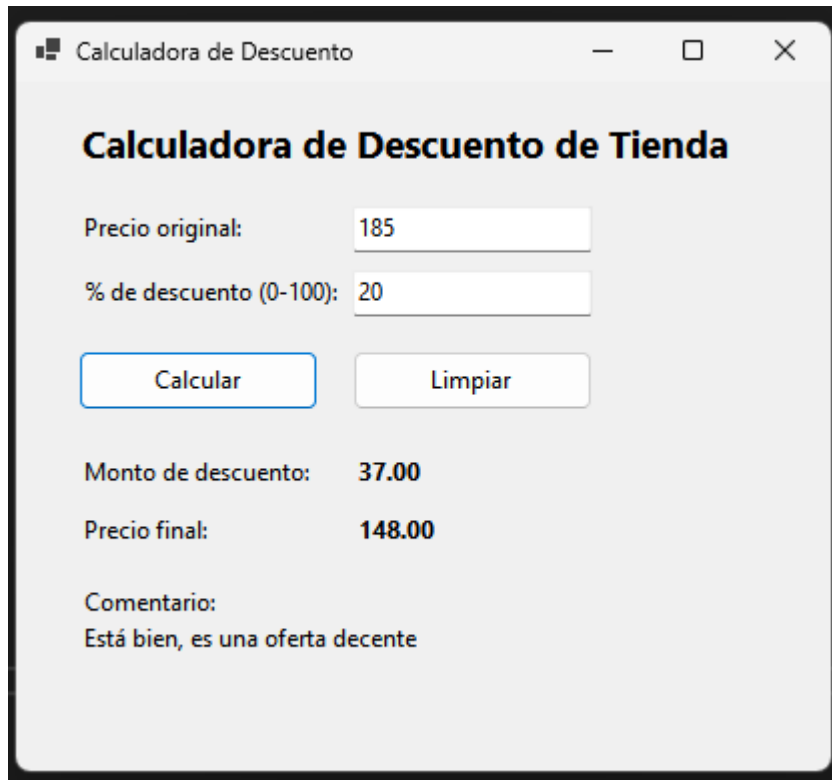
    private class TareaItem
    {
        public string Texto { get; set; } = string.Empty;
        public bool Completada { get; set; }
    }
}

```

Nota: En Windows Forms, el diseño visual se guarda en archivos Designer.cs generados automáticamente por el IDE. En esta entrega se incluye el código del formulario con los eventos y el archivo Program.cs como punto de entrada.

## Programa 07: Calculadora de descuento en tienda

Aplicación Windows Forms que calcula el monto descontado y el precio final a partir de un precio original y un porcentaje de descuento.



**Calculadora de Descuento de Tienda**

Precio original: 185

% de descuento (0-100): 20

Calcular Limpiar

Monto de descuento: 37.00

Precio final: 148.00

Comentario:  
Está bien, es una oferta decente

### Código utilizado

```
using System;
using System.Windows.Forms;

namespace CalculadoraDescuentoTienda;

internal static class Program
{
    /// <summary>
    /// Punto de entrada principal para la aplicación.
    /// </summary>
    [STAThread]
    static void Main()
    {
        ApplicationConfiguration.Initialize();
        Application.Run(new MainForm());
    }
}
```

Archivo del formulario (código de eventos):

```
using System;
using System.Windows.Forms;

namespace CalculadoraDescuentoTienda
{
    public partial class MainForm : Form
    {
        public MainForm()
        {
            InitializeComponent();
        }
    }
}
```

```

private void btnCalcular_Click(object sender, EventArgs e)
{
    string textoPrecio = txtPrecio.Text.Trim();
    string textoDescuento = txtDescuento.Text.Trim();

    if (string.IsNullOrEmpty(textoPrecio))
    {
        MessageBox.Show("Ingrese el precio original.", "Dato faltante",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        txtPrecio.Focus();
        return;
    }

    if (string.IsNullOrEmpty(textoDescuento))
    {
        MessageBox.Show("Ingrese el porcentaje de descuento.", "Dato
faltante",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        txtDescuento.Focus();
        return;
    }

    if (!double.TryParse(textoPrecio, out double precio) || precio <= 0)
    {
        MessageBox.Show("Ingrese un precio válido mayor que 0.", "Dato
inválido",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        txtPrecio.Focus();
        return;
    }

    if (!double.TryParse(textoDescuento, out double porcentajeDescuento))
    {
        MessageBox.Show("Ingrese un porcentaje de descuento válido.", "Dato
inválido",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        txtDescuento.Focus();
        return;
    }

    if (porcentajeDescuento < 0 || porcentajeDescuento > 100)
    {
        MessageBox.Show("El porcentaje de descuento debe estar entre 0 y
100.", "Dato inválido",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        txtDescuento.Focus();
        return;
    }

    double montoDescuento = precio * (porcentajeDescuento / 100.0);
    double precioFinal = precio - montoDescuento;

    lblMontoDescuentoValor.Text = montoDescuento.ToString("0.00");
    lblPrecioFinalValor.Text = precioFinal.ToString("0.00");

    lblComentario.Text = GenerarComentario(porcentajeDescuento);
}

private string GenerarComentario(double porcentaje)
{
    if (porcentaje == 0)
    {
        return "No hay oferta, es el precio normal";
    }
    else if (porcentaje > 0 && porcentaje <= 10)

```

```

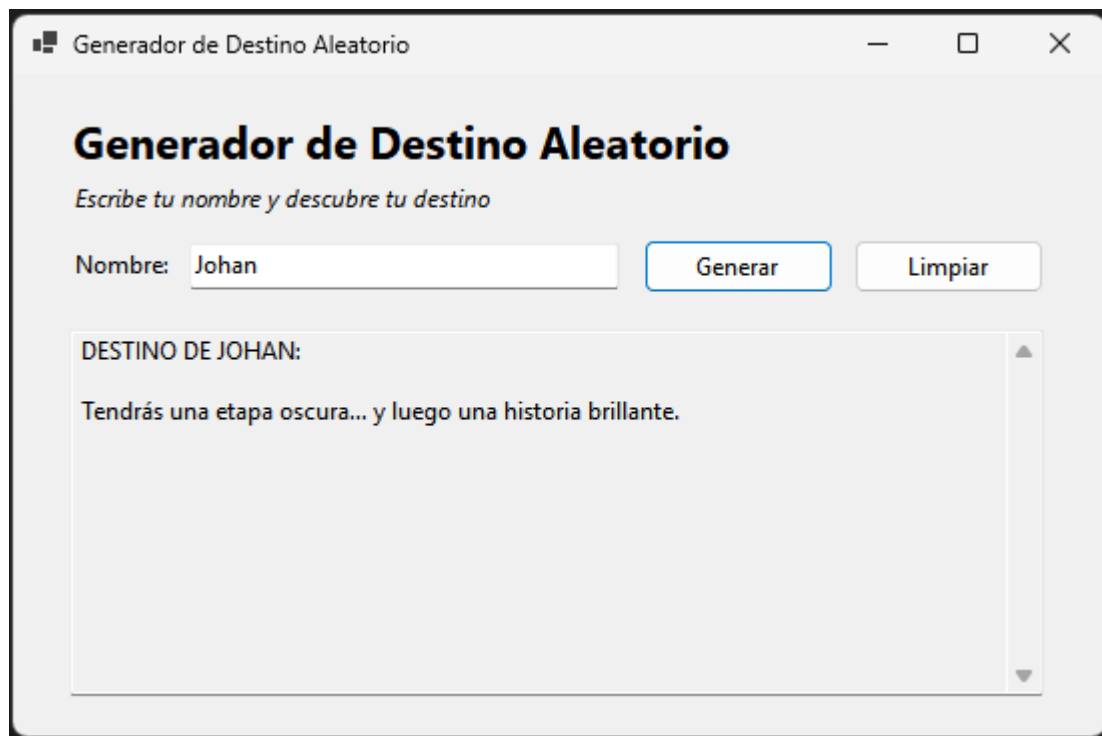
        {
            return "mal descuento.";
        }
        else if (porcentaje > 10 && porcentaje <= 30)
        {
            return "Está bien, es una oferta decente";
        }
        else if (porcentaje > 30 && porcentaje <= 60)
        {
            return "Buen descuento, aprovecha antes de que se arrepientan";
        }
        else if (porcentaje > 60 && porcentaje < 100)
        {
            return "¡Tremenda oferta!";
        }
        else // 100
        {
            return "¿Gratis? sospechoso";
        }
    }

private void btnLimpiar_Click(object sender, EventArgs e)
{
    txtPrecio.Clear();
    txtDescuento.Clear();
    lblMontoDescuentoValor.Text = "0.00";
    lblPrecioFinalValor.Text = "0.00";
    lblComentario.Text = string.Empty;
    txtPrecio.Focus();
}
}
}

```

## Programa 08: Generador de destino aleatorio

Aplicación Windows Forms que asigna un 'destino' aleatorio a un nombre (mantiene el mismo destino si se repite el nombre).



### Código utilizado

```
using System;
using System.Windows.Forms;

namespace GeneradorDestinoAleatorio;

internal static class Program
{
    [STAThread]
    static void Main()
    {
        ApplicationConfiguration.Initialize();
        Application.Run(new MainForm());
    }
}

Archivo del formulario (código de eventos):
using System;
using System.Collections.Generic;
using System.Windows.Forms;

namespace GeneradorDestinoAleatorio
{
    public partial class MainForm : Form
    {
        private readonly Random _random = new();
        private readonly Dictionary<string, string> _destinosPorNombre =
new(StringComparer.OrdinalIgnoreCase);

        private readonly List<string> _destinos = new()
        {
            "Serás MILLONARIO... pero no por herencia, sino porque crearás algo que
nadie más se atrevió a intentar.",
```

```

        "Serás famoso sin querer, por un video que subirás de broma y hará reír
a medio internet.",
        "Serás youtuber y al inicio nadie te verá... hasta que un clip se vuelva
viral y cambie todo.",
        "Serás tiktoker accidental: subirás un video aburrido y millones dirán
que eres leyenda.",
        "Serás pobre... pero solo porque gastarás todo viajando y viviendo
historias que nadie tendrá.",
        "Engordarás... pero será porque comerás demasiado rico en cada país que
visites.",
        "Te volverás famoso por algo absurdo... y luego te tomarán en serio por
tu talento.",
        "Serás rico en contactos, aunque no siempre en dinero.",
        "Fracasarás fuerte una vez... y después todos dirán que siempre creyeron
en ti.",
        "Tendrás haters sin haberlos buscado... señal de que hiciste algo
importante.",
        "Serás conocido por resolver problemas que nadie más quiere tocar.",
        "Te harás viral por decir una verdad incómoda.",
        "Vivirás una etapa donde nadie te entiende... hasta que de repente todos
te copien.",
        "Ganarás dinero con algo que hoy te parece una pérdida de tiempo.",
        "Serás recordado por ayudar cuando nadie más quiso.",
        "Harás algo tan raro que un día dirás: 'menos mal no hice caso'. ",
        "Serás el ejemplo de alguien que parecía normal... y terminó siendo
leyenda.",
        "Un día verás tu nombre y dirás: 'No sabía que iba a llegar tan lejos'.
",
        "Tendrás una caída fuerte que te hará rediseñar toda tu vida.",
        "Te volverás famoso en un lugar donde nadie te conocía.",
        "Ganarás dinero por explicar algo mejor que los demás.",
        "La gente pensará que tuviste suerte... tú sabrás que fue insistencia.",
        "Te volverás experto en algo que hoy apenas entiendes.",
        "Serás viral por algo que ni siquiera planeaste.",
        "Harás dinero mientras otros aún creen que es imposible.",
        "Serás pobre una temporada... pero aprenderás lo que muchos ricos nunca
sabrán.",
        "Un día dirás: 'valió la pena cada cosa rara que hice'. ",
        "Serás famoso por ayudar más que por presumir.",
        "Cambiarás de rumbo tantas veces que pensarán que estás perdido... hasta
que llegues primero.",
        "Tu éxito llegará callado y se volverá escandaloso.",
        "Serás conocido por hacer cosas imposibles parecer fáciles.",
        "Un error tuyo abrirá una puerta enorme.",
        "Serás más grande en la vida de otros que en internet.",
        "Te volverás referente sin buscarlo.",
        "Te reirás del destino que hoy te dieron.",
        "Serás peligroso... pero para la mediocridad.",
        "Tendrás una etapa oscura... y luego una historia brillante.",
        "Serás diferente, y eso te salvará.",
        "Un día alguien dirá: 'yo lo conocí cuando no era nadie'. ",
        "Cambiarás la vida de alguien sin darte cuenta.",
        "Serás millonario en historias.",
        "Fracasarás mejor que otros triunfan.",
        "Serás inspiración para quien hoy ni te nota.",
        "Te volverás viral cuando ya no te importe serlo."
    };

    public MainForm()
    {
        InitializeComponent();
    }

    private void btnGenerar_Click(object sender, EventArgs e)
    {
        string nombre = txtNombre.Text.Trim();

```

```

        if (string.IsNullOrEmpty(nombre))
        {
            MessageBox.Show("Escribe un nombre antes de generar el destino.",
"Falta nombre", MessageBoxButtons.OK, MessageBoxIcon.Information);
            return;
        }

        if (!_destinosPorNombre.ContainsKey(nombre))
        {
            _destinosPorNombre[nombre] =
_destinos[_random.Next(_destinos.Count)];
        }

        txtDestino.Text = "DESTINO DE " + nombre.ToUpper() + ":

" + _destinosPorNombre[nombre];
    }

    private void btnLimpiar_Click(object sender, EventArgs e)
    {
        txtNombre.Clear();
        txtDestino.Clear();
        txtNombre.Focus();
    }
}

```

## Programa 09: Test de arquetipos Soldado - Poeta - Rey

Aplicación Windows Forms que calcula un resultado de arquetipo basado en respuestas de radio buttons.

### Código utilizado

```
using System;
using System.Windows.Forms;
```

```
namespace SoldadoPoetaReySimulado;
```

```
internal static class Program
{
    /// <summary>
    /// Punto de entrada principal para la aplicación.
    /// </summary>
    [STAThread]
    static void Main()
    {
        ApplicationConfiguration.Initialize();
        Application.Run(new MainForm());
    }
}
```

Archivo del formulario (código de eventos):

```
using System;
using System.Windows.Forms;
```

```
namespace SoldadoPoetaReySimulado
```

```
{
    public partial class MainForm : Form
    {
        public MainForm()
        {
            InitializeComponent();
        }

        private void btnCalcular_Click(object sender, EventArgs e)
        {
            int soldado = 0;
            int poeta = 0;
            int rey = 0;

            // Pregunta 1
            if (rb1Soldado.Checked) soldado++;
        }
    }
}
```

```

if (rb1Poeta.Checked) poeta++;
if (rb1Rey.Checked) rey++;

// Pregunta 2
if (rb2Soldado.Checked) soldado++;
if (rb2Poeta.Checked) poeta++;
if (rb2Rey.Checked) rey++;

// Pregunta 3
if (rb3Soldado.Checked) soldado++;
if (rb3Poeta.Checked) poeta++;
if (rb3Rey.Checked) rey++;

// Pregunta 4
if (rb4Soldado.Checked) soldado++;
if (rb4Poeta.Checked) poeta++;
if (rb4Rey.Checked) rey++;

// Pregunta 5
if (rb5Soldado.Checked) soldado++;
if (rb5Poeta.Checked) poeta++;
if (rb5Rey.Checked) rey++;

int totalRespondidas = soldado + poeta + rey;

if (totalRespondidas == 0)
{
    MessageBox.Show("Responde al menos una pregunta antes de
calcular.", "Sin datos",
        MessageBoxButtons.OK, MessageBoxIcon.Information);
    return;
}

string arquetipoPrincipal;
string descripcion;

if (soldado > poeta && soldado > rey)
{
    arquetipoPrincipal = "SOLDADO";
    descripcion = "Tiendes a ser alguien de acción. Prefieres hacer que
hablar, asumir responsabilidades y sacar adelante las cosas con disciplina.";
}
else if (poeta > soldado && poeta > rey)
{
    arquetipoPrincipal = "POETA";
    descripcion = "Tu fortaleza está en las ideas, la creatividad y la
sensibilidad. Piensas en significados, historias y emociones detrás de lo que
haces.";
}
else if (rey > soldado && rey > poeta)
{
    arquetipoPrincipal = "REY";
    descripcion = "Tienes una mente estratégica y organizadora. Te
gusta ver el panorama completo, decidir rumbos y coordinar a otras personas.";
}
else
{
    arquetipoPrincipal = "COMBINADO";
    descripcion = "Tus respuestas muestran una mezcla equilibrada de
Soldado, Poeta y Rey. No encajas en una sola etiqueta, sino que integras varios
estilos.";
}

txtResultado.Text = $"RESULTADO DEL TEST" + Environment.NewLine +
    "-----" +
Environment.NewLine +

```

```

+
Environment.NewLine +
        $"Puntaje Soldado: {soldado}" + Environment.NewLine
        $"Puntaje Poeta: {poeta}" + Environment.NewLine +
        $"Puntaje Rey: {rey}" + Environment.NewLine +
        Environment.NewLine +
        $"Arquetipo principal: {arquetipoPrincipal}" +
        Environment.NewLine +
        descripcion + Environment.NewLine +
        Environment.NewLine +
        "1";
    }

    private void btnLimpiar_Click(object sender, EventArgs e)
    {
        foreach (Control c in Controls)
        {
            LimpiarRadios(c);
        }

        txtResultado.Clear();
    }

    private void LimpiarRadios(Control control)
    {
        if (control is RadioButton rb)
        {
            rb.Checked = false;
        }

        foreach (Control child in control.Controls)
        {
            LimpiarRadios(child);
        }
    }

    private void rb4Poeta_CheckedChanged(object sender, EventArgs e)
    {
    }

    private void txtResultado_TextChanged(object sender, EventArgs e)
    {
    }

    private void lblTitulo_Click(object sender, EventArgs e)
    {
    }

    private void MainForm_Load(object sender, EventArgs e)
    {
    }
}

```

## Programa 10: Simulador de vida diaria

Aplicación Windows Forms que analiza horas de estudio, pantallas, ejercicio y sueño, mostrando diagnóstico y una barra de progreso del día.

**Simulador de Vida Diaria**  
*Distribuye tus 24 horas y analiza tus hábitos diarios*

Horas de estudio: 3  
Horas de pantallas/ocio: 1  
Horas de ejercicio: 1  
Horas de sueño: 7

Analizar rutina    Limpiar

Total de horas: 12.0

Porcentaje de uso del día (0-24h)

**SIMULADOR DE VIDA DIARIA**  
Horas de estudio: 3.0  
Horas de pantallas/ocio: 1.0  
Horas de ejercicio: 1.0  
Horas de sueño: 7.0  
Total de horas declaradas: 12.0

Buen hábito de sueño. Mantén ese rango de horas para descansar bien.  
Tienes un buen nivel de actividad física diaria.  
Mantienes un uso moderado de pantallas, eso ayuda a tu concentración.  
Buena cantidad de horas de estudio, sigue así.

Resumen general: Tu rutina diaria es bastante equilibrada. Buen trabajo.

### Código utilizado

```
using System;
using System.Windows.Forms;

namespace SimuladorVidaDiaria;

internal static class Program
{
    /// <summary>
    /// Punto de entrada principal para la aplicación.
    /// </summary>
    [STAThread]
    static void Main()
    {
        ApplicationConfiguration.Initialize();
        Application.Run(new MainForm());
    }
}

Archivo del formulario (código de eventos):
using System;
using System.Drawing;
using System.Windows.Forms;

namespace SimuladorVidaDiaria
{
    public partial class MainForm : Form
    {
        public MainForm()
        {
            InitializeComponent();
        }
    }
}
```

```

private void btnAnalizar_Click(object sender, EventArgs e)
{
    if (!ValidarYCalcular(out double hEstudio, out double hPantalla, out
double hEjercicio, out double hSueno))
    {
        return;
    }

    double total = hEstudio + hPantalla + hEjercicio + hSueno;
    lblTotalHorasValor.Text = total.ToString("0.0");

    // Actualizar barra de progreso (0 a 24 horas)
    int progreso = (int)Math.Round((total / 24.0) * 100);
    if (progreso < 0) progreso = 0;
    if (progreso > 100) progreso = 100;
    progressBarDia.Value = progreso;

    // Ahora no usamos string, sino que escribimos directo en el
RichTextBox
    GenerarDiagnostico(hEstudio, hPantalla, hEjercicio, hSueno, total);
}

private bool ValidarYCalcular(out double hEstudio, out double hPantalla,
out double hEjercicio, out double hSueno)
{
    hEstudio = hPantalla = hEjercicio = hSueno = 0;

    if (!double.TryParse(txtEstudio.Text, out hEstudio) ||
        !double.TryParse(txtPantalla.Text, out hPantalla) ||
        !double.TryParse(txtEjercicio.Text, out hEjercicio) ||
        !double.TryParse(txtSueno.Text, out hSueno))
    {
        MessageBox.Show("Por favor ingresa solo números válidos (pueden ser
decimales) en todos los campos.",
                        "Datos inválidos", MessageBoxButtons.OK,
MessageBoxIcon.Warning);
        return false;
    }

    if (hEstudio < 0 || hPantalla < 0 || hEjercicio < 0 || hSueno < 0)
    {
        MessageBox.Show("Las horas no pueden ser negativas.", "Dato
inválido",
                        MessageBoxButtons.OK, MessageBoxIcon.Warning);
        return false;
    }

    double total = hEstudio + hPantalla + hEjercicio + hSueno;
    if (total > 24.5)
    {
        MessageBox.Show("La suma de horas supera las 24 horas del día.
Revisa tus datos.",
                        "Revisa la distribución", MessageBoxButtons.OK,
MessageBoxIcon.Warning);
        return false;
    }

    return true;
}

// Escribe una línea en verde (bueno) o rojo (malo)
private void Linea(string texto, bool esBueno)
{
    txtResultado.SelectionStart = txtResultado.TextLength;
    txtResultado.SelectionLength = 0;
}

```

```

        txtResultado.SelectionColor = esBueno ? Color.Green : Color.Red;
        txtResultado.AppendText(texto + Environment.NewLine);
        txtResultado.SelectionColor = Color.Black;
    }

    // Escribe un título en azul
    private void Titulo(string texto)
    {
        txtResultado.SelectionStart = txtResultado.TextLength;
        txtResultado.SelectionColor = Color.DarkBlue;
        txtResultado.AppendText(texto + Environment.NewLine);
        txtResultado.SelectionColor = Color.Black;
    }

    private void GenerarDiagnostico(double hEstudio, double hPantalla, double
hEjercicio, double hSueno, double total)
    {
        txtResultado.Clear();

        // Encabezado
        Titulo("SIMULADOR DE VIDA DIARIA");
        Titulo("-----");

        txtResultado.AppendText($"Horas de estudio:
{hEstudio:0.0}{Environment.NewLine}");
        txtResultado.AppendText($"Horas de pantallas/ocio:
{hPantalla:0.0}{Environment.NewLine}");
        txtResultado.AppendText($"Horas de ejercicio:
{hEjercicio:0.0}{Environment.NewLine}");
        txtResultado.AppendText($"Horas de sueño:
{hSueno:0.0}{Environment.NewLine}");
        txtResultado.AppendText($"Total de horas declaradas:
{total:0.0}{Environment.NewLine}{Environment.NewLine}");

        // SUEÑO
        if (hSueno < 6)
        {
            Linea("Duermes poco. Intenta dormir al menos entre 7 y 8 horas para
cuidar tu salud.", false);
        }
        else if (hSueno >= 7 && hSueno <= 9)
        {
            Linea("Buen hábito de sueño. Mantén ese rango de horas para
descansar bien.", true);
        }
        else if (hSueno > 10)
        {
            Linea("Duermes demasiado. Podría estar afectando el tiempo para
otras actividades importantes.", false);
        }

        // EJERCICIO
        if (hEjercicio == 0)
        {
            Linea("No realizas ejercicio. Intenta dedicar al menos 30 minutos
al día a moverte.", false);
        }
        else if (hEjercicio >= 0.5 && hEjercicio <= 2)
        {
            Linea("Tienes un buen nivel de actividad física diaria.", true);
        }
        else if (hEjercicio > 3)
        {
            Linea("Mucho ejercicio en un solo día puede cansarte demasiado si
no lo equilibras.", false);
        }
    }

```

```

        // PANTALLA
        if (hPantalla > 6)
        {
            Linea("Pasas demasiado tiempo en pantallas. Intenta reducirlo o
hacer pausas activas.", false);
        }
        else if (hPantalla <= 3)
        {
            Linea("Mantienes un uso moderado de pantallas, eso ayuda a tu
concentración.", true);
        }

        // ESTUDIO
        if (hEstudio < 2)
        {
            Linea("Pocas horas de estudio. Si eres estudiante, podrías
necesitar más tiempo para tus cursos.", false);
        }
        else if (hEstudio >= 3 && hEstudio <= 6)
        {
            Linea("Buena cantidad de horas de estudio, sigue así.", true);
        }
        else if (hEstudio > 8)
        {
            Linea("Estudias muchas horas. Cuidado con el agotamiento, también
necesitas descanso.", false);
        }

        txtResultado.AppendText(Environment.NewLine + "Resumen general: ");

        // RESUMEN: escribimos también con color
        if (hSueno >= 7 && hSueno <= 9 &&
            hEjercicio >= 0.5 && hEjercicio <= 2 &&
            hPantalla <= 4 &&
            hEstudio >= 2 &&
            total <= 24.2)
        {
            Linea("Tu rutina diaria es bastante equilibrada. Buen trabajo.",
true);
        }
        else if (total <= 24.2)
        {
            Linea("Tu rutina tiene aspectos positivos, pero puedes mejorar el
equilibrio entre estudio, descanso, ejercicio y ocio.", false);
        }
        else
        {
            Linea("Los tiempos no son realistas para un día de 24 horas. Revisa
tu planificación.", false);
        }
    }

    private void btnLimpiar_Click(object sender, EventArgs e)
    {
        txtEstudio.Clear();
        txtPantalla.Clear();
        txtEjercicio.Clear();
        txtSueno.Clear();
        txtResultado.Clear();
        lblTotalHorasValor.Text = "0";
        progressBarDia.Value = 0;
        txtEstudio.Focus();
    }

    private void txtResultado_TextChanged(object sender, EventArgs e)

```

```
    {  
        // Puedes dejarlo vacío, lo exige el diseñador por el evento conectado.  
    }  
}
```