

EXERCÍCIO 1 –

```
using System;

class Livro
{
    private string titulo;
    private string autor;
    private int anoPublicacao;

    public Livro()
    {
        titulo = "Desconhecido";
        autor = "Anônimo";
        anoPublicacao = 0;
    }

    public Livro(string titulo, string autor, int anoPublicacao)
    {
        this.titulo = titulo;
        this.autor = autor;
        this.anoPublicacao = anoPublicacao;
    }

    public void ExibirDetalhes()
    {
        Console.WriteLine($"Detalhes: título: {titulo}, autor: {autor}, ano de publicação: {anoPublicacao}");
    }
}

class Program
{
    static void Main()
    {
        Console.Clear();
        Livro livro1 = new Livro();
        livro1.ExibirDetalhes();

        Livro livro2 = new Livro("1984", "George Orwell", 1949);
        livro2.ExibirDetalhes();
    }
}
```

EXERCÍCIO 2 –

```
using System;

class Carro
{
    public string Modelo { get; set; }
    public string Marca { get; set; }
    public int Ano { get; set; }

    public Carro()
    {
    }

    public Carro(string modelo, string marca, int ano)
    {
        Modelo = modelo;
        Marca = marca;
        Ano = ano;
    }

    public Carro(Carro outroCarro)
    {
        Modelo = outroCarro.Modelo;
        Marca = outroCarro.Marca;
        Ano = outroCarro.Ano;
    }

    ~Carro()
    {
        Console.WriteLine($"Destrutor chamado. O carro {Modelo} está sendo removido da memória.");
    }
}

class Program
{
    static void Main(string[] args)
    {
        Carro carro1 = new Carro("Uno", "Fiat", 1980);
        Carro carro2 = new Carro(carro1);

        Console.WriteLine($"Carro 1: {carro1.Modelo}, {carro1.Marca}, {carro1.Ano}");
        Console.WriteLine($"Carro 2 (cópia): {carro2.Modelo}, {carro2.Marca}, {carro2.Ano}");

        GC.Collect();
        GC.WaitForPendingFinalizers();
    }
}
```

EXERCÍCIO 3 –

```
using System;

public class Pessoa
{
    public string Nome { get; set; }
    public int Idade { get; set; }

    public Pessoa() : this("Sem nome", 0)
    {
        Console.WriteLine("Construtor padrão chamado. Objeto criado.");
    }

    public Pessoa(string nome, int idade)
    {
        Nome = nome;
        Idade = idade;
        Console.WriteLine($"Construtor com parâmetros chamado. {Nome} criado(a).");
    }

    public Pessoa(Pessoa outraPessoa) : this(outraPessoa.Nome, outraPessoa.Idade)
    {
        Console.WriteLine($"Construtor de cópia chamado. Cópia de {Nome} criada.");
    }

    ~Pessoa()
    {
        Console.WriteLine($"Destrutor chamado. {Nome} está sendo removido da memória.");
    }

    public void ExibirInfo()
    {
        Console.WriteLine($"Nome: {Nome}, Idade: {Idade}");
    }
}

class Program
{
    static void Main()
    {
        Pessoa p1 = new Pessoa();
        Pessoa p2 = new Pessoa("Alice", 30);
        Pessoa p3 = new Pessoa(p2);

        p1.ExibirInfo();
        p2.ExibirInfo();
        p3.ExibirInfo();

        GC.Collect();
        GC.WaitForPendingFinalizers();
    }
}
```

EXERCÍCIO 4 –

```
using System;

public class Contador
{
    public static int TotalInstancias = 0;

    public Contador()
    {
        TotalInstancias++;
        Console.WriteLine($"Instância criada. Total: {TotalInstancias}");
    }

    ~Contador()
    {
        TotalInstancias--;
        Console.WriteLine($"Instância destruída. Total: {TotalInstancias}");
    }

    public static void MostrarTotal()
    {
        Console.WriteLine($"Total de instâncias ativas: {TotalInstancias}");
    }
}

class Program
{
    static void Main()
    {
        Console.WriteLine("Criando instâncias:");
        Contador a = new Contador();
        Contador b = new Contador();
        Contador.MostrarTotal();

        Console.WriteLine("\nForçando coleta de lixo:");
        a = null;
        b = null;
        GC.Collect();
        GC.WaitForPendingFinalizers();

        Contador.MostrarTotal();
    }
}
```

EXERCÍCIO 5 –

```
using System;

class ArquivoLogger
{
    public ArquivoLogger()
    {
        Console.WriteLine("Arquivo aberto.");
    }

    public void Log(string mensagem)
    {
        Console.WriteLine($"Log: {mensagem}");
    }

    ~ArquivoLogger()
    {
        Console.WriteLine("Arquivo fechado (Finalizador).");
    }
}

class Program
{
    static void Main()
    {
        var logger1 = new ArquivoLogger();
        logger1.Log("Mensagem 1");
        logger1.Log("Mensagem 2");

        Console.WriteLine("\nCriando outro logger:");
        var logger2 = new ArquivoLogger();
        logger2.Log("Mensagem 3");

        Console.WriteLine("\nForçando coleta de lixo...");
        GC.Collect();
        GC.WaitForPendingFinalizers();

        Console.WriteLine("\nFim do programa.");
    }
}
```