

Algoritmos e Técnicas de Programação - Aula 10

Aluno: **Jean Carlo Machado Amaral**

```
using System;
public class Console1
{
    public static void Main(String[] args)
    {
        int[] conjuntoA = lerConjuntos();
        int[] conjuntoB = lerConjuntos();
        int[] conjuntoC = ordenar(conjuntoA, conjuntoB);

        Console.Write(" Conjunto C: ");
        for (int i = 0; i < conjuntoC.Length; i++)
        {
            Console.Write(conjuntoC[i] + " ");
        }
    }

    public static int[] lerConjuntos()
    {
        Console.Write("Informe o número de elementos do conjunto: ");
        int n = int.Parse(Console.ReadLine());
        int[] conjunto = new int[n];
        Console.Write("Informe os elementos do conjunto: ");

        for (int i = 0; i < n; i++)
        {
            conjunto[i] = int.Parse(Console.ReadLine());
        }
        return conjunto;
    }

    public static int[] ordenar(int[] conjuntoA, int[] conjuntoB)
    {
        int sizeC = conjuntoA.Length + conjuntoB.Length;
        int[] conjuntoC = new int[sizeC];

        int indiceA = 0;
        int indiceB = 0;
        int indiceC = 0;

        while (indiceA < conjuntoA.Length && indiceB < conjuntoB.Length)
        {
            if (conjuntoA[indiceA] <= conjuntoB[indiceB])
            {
                conjuntoC[indiceC] = conjuntoA[indiceA];
                indiceA++;
            }
        }
    }
}
```

```

    }
    else
    {
        conjuntoC[indiceC] = conjuntoB[indiceB];
        indiceB++;
    }
    indiceC++;
}

while (indiceA < conjuntoA.Length)
{
    conjuntoC[indiceC] = conjuntoA[indiceA];
    indiceA++;
    indiceC++;
}

while (indiceB < conjuntoB.Length)
{
    conjuntoC[indiceC] = conjuntoB[indiceB];
    indiceB++;
    indiceC++;
}

return conjuntoC;
}
}

```

QUESTÃO 2

```

using System;
public class Program2
{
    public static void Main(string[] args)
    {
        int[] atp = leitura("Algoritmos e Técnicas de Programação");
        int[] calculo = leitura("Cálculo");
        // intersecao == presente em dois conjuntos
        int[] intersecao = vetorintersecao(atp, calculo);

        if(intersecao.Length == 0)
        {
            //verifica se possui algum elemento no vetor
            Console.WriteLine("Não existe alunos cursando os dois cursos ao mesmo tempo");
        }
        else
        {
            //se houver printa

```

```

        Console.WriteLine(" Alunos matriculados nas duas disciplinas: ");
        for (int i = 0; i < intersecao.Length; i++)
        {
            Console.Write(" " + intersecao[i]);
        }
    }

}

public static int[] leitura(string disciplina)
{
    int[] matricula = new int[10];
    Console.Write("Matriculados em: \n" + disciplina + "\n");
    for (int i = 0; i < 10; i++)
    {
        matricula[i] = int.Parse(Console.ReadLine());
    }
    return matricula;
}

public static int[] vetorintersecao(int[] vetor1, int[]vetor2)
{
    //define tamanho do vetor temporario
    int[] temporario = new int[Math.Min(vetor1.Length, vetor2.Length)];
    int contador = 0;

    for(int i=0; i<vetor1.Length;i++)
    {
        int matricula = vetor1[i];
        //booleano feito para verificar se a matricula esta presente no vetor2
        bool existe = false;

        //loop para percorrer o vetor2
        for (int j = 0; j < vetor2.Length; j++)
        {
            //verifica: index é igual a matricula
            if (vetor2[j] == matricula)
            {
                //se for verdade para no break
                existe = true;
                break;
            }
        }
        if (existe)
        {
            temporario[contador] = matricula;
            contador++;
            if(contador == temporario.Length)
            {
                break;
            }
        }
    }
}

```

```

        //Para quando exceder o tamanho do vetor
    }
}
}

int[] intersecao = new int[contador];
//copia os valores de temporario para intersecao
for (int k = 0; k < contador; k++)
{
    intersecao[k] = temporario[k];
}
//retorna como resultado
return intersecao;

}
}

```

Questao 3

```

using System;
public class Program3
{
    public static void Main(string[] args)
    {

        int[] usuario = insereValor(10);
        int[] numeros = gerarNumero(6);
        int acertos = verificacao(numeros, usuario);
        Console.WriteLine("\n Número de acertos: " + acertos);

    }
    public static int[] gerarNumero(int tamanho)
    {
        int[] megasena = new int[tamanho];
        Random rd = new Random();
        for (int i = 0; i < 6; i++)
        {
            //vai gerar numeros ate 50 pra ficar mais facil de acertar
            megasena[i] = rd.Next(0, 51);
        }

        Console.WriteLine("\n Números gerados:");
        for (int i = 0; i < 6; i++)
        {
            Console.Write(megasena[i] + " ");
        }

        return megasena;
    }
}

```

```

    }
    public static int[] insereValor(int valor)
    {
        Console.WriteLine();
        int [] valores = new int[valor];
        Console.Write("Digite os valores: ");
        for (int i = 0; i < 10; i++)
        {
            valores[i] = int.Parse(Console.ReadLine());
        }

        Console.WriteLine("Valores digitados: \n");
        for (int i = 0; i < 10; i++)
        {
            Console.Write(" " + valores[i]);
        }
        Console.WriteLine();
        return valores;
    }

    public static int verificacao(int[] megasena, int [] valores) {
        int contador = 0;

        for (int i = 0; i < megasena.Length; i++)
        {
            for (int j = 0; j < valores.Length; j++)
            {
                if (valores[j] == megasena[i])
                {
                    contador++;
                }
            }
        }
        if(contador == 0)
        {
            Console.WriteLine("\n Não acertou nenhum número");
        }
        return contador;
    }
}

```

Questão 4

```

// questao 4
//0 pedra 1 papel 2 tesoura 3 empate
using System;
public class Program4

```

```

{
    public static void Main(string[] args)
    {
        int opcao = lerOpcao();
        int computador = sorteia();
        int resultado = verificaGanhador(opcao, computador);

    }
    public static int lerOpcao()
    {
        int escolha;
        Console.WriteLine("Escolha uma opção: 0 pedra 1 papel 2 tesoura ");
        escolha = int.Parse(Console.ReadLine());
        Console.WriteLine($"Você escolheu: {escolha}");
        return escolha;

    }
    public static int sorteia()
    {
        int escolhaComputador;
        Random rd = new Random();
        escolhaComputador = rd.Next(0,3);
        Console.WriteLine();
        Console.WriteLine($"Vez do computador: {escolhaComputador}");
        return escolhaComputador;
    }
    public static int verificaGanhador(int opcao, int computador)
    {
        //0 pedra 1 papel 2 tesoura
        //1 jogador 2 computador 3 empate
        Console.WriteLine();
        int verifica=0;
        if( computador == 0 && opcao ==1)
        {
            verifica = 1;
            Console.WriteLine($"Você ganhou :) \n status: {verifica}");
        }
        if(computador ==0 && opcao ==2)
        {
            verifica = 2;
            Console.WriteLine($"Você perdeu :( \n status: {verifica}");
        }
        if(computador ==0 && opcao==0)
        {
            verifica=3;
            Console.WriteLine($"Empatou \n status: {verifica}");
        }
    }
}

```

```

if(computador ==1 && opcao == 0)
{
    verifica = 2;
    Console.WriteLine($"Você perdeu :( \n status: {verifica}");

}
if (computador == 1 && opcao == 1)
{
    verifica = 3;
    Console.WriteLine($"Empatou \n status: {verifica}");
}
if (computador == 1 && opcao == 2)
{
    verifica = 1;
    Console.WriteLine($"Você ganhou :) \n status: {verifica}");
}
if(computador == 2 && opcao == 0)
{
    verifica = 1;
    Console.WriteLine($"Você ganhou :) \n status: {verifica}");
}
if(computador == 2 && opcao == 1)
{
    verifica = 2;
    Console.WriteLine($"Você perdeu :( \n status: {verifica}");
}
if (computador==2 && opcao == 2)
{
    verifica = 3;
    Console.WriteLine($"Empatou \n status: {verifica}");

}

    return verifica;
}
}

```