

DOCUMENTAÇÃO DETALHADA DOS EXERCÍCIOS - C#

EXERCÍCIO 3: CÁLCULO DE FATORIAL COM WHILE

1. Objetivo:

Solicitar ao usuário um número inteiro positivo e calcular seu fatorial usando a estrutura de repetição 'while'.

2. Passo a Passo Lógico:

- Solicita que o usuário digite um número inteiro positivo.
- Inicializa uma variável 'fatorial' com valor 1.
- Enquanto o número for maior que 1:
- Multiplica o valor atual de 'fatorial' pelo número.
- Decrementa o número.
- Ao final, exibe o valor do fatorial calculado.

3. Estruturas e Comandos Utilizados:

- `Console.WriteLine()` e `Console.ReadLine()` para interação com o usuário.
- `int.Parse()` para converter string para inteiro.
- Estrutura de repetição `while`.
- Operadores de atribuição (`*`) e decremento (`--`).

4. Justificativa:

O uso do `while` é ideal nesse caso porque o número de repetições depende do valor inserido pelo usuário. A lógica é simples e eficaz para multiplicar valores sequencialmente de cima para baixo (ex: $5 * 4 * 3 * 2 * 1$).

Código:

```
```csharp
```

```
using System;
```

```
class Program
```

```
{
```

```
static void Main()
```

```
{
```

```
Console.Write("Digite um número inteiro positivo: ");
```

```
int numero = int.Parse(Console.ReadLine());
```

```
int fatorial = 1;
```

```
while (numero > 1)
```

```
{
```

```
fatorial *= numero;
```

```
numero--;
```

```
}
```

```
Console.WriteLine($"Fatorial = {fatorial}");
```

```
}
```

```
}
```

```
```
```

```
---
```

EXERCÍCIO 9: JOGO DA ADIVINHAÇÃO

1. Objetivo:

Criar um jogo onde o programa sorteia um número aleatório de 1 a 100 e o usuário deve adivinhar, recebendo dicas após cada tentativa.

2. Passo a Passo Lógico:

- O programa gera um número aleatório entre 1 e 100.
- Pede que o usuário tente adivinhar.
- A cada tentativa:
 - Verifica se o número digitado é maior, menor ou igual ao número sorteado.
 - Informa ao usuário se ele deve tentar mais alto ou mais baixo.
 - Conta quantas tentativas foram feitas.
- O loop continua até o número correto ser adivinhado.
- Ao acertar, informa o número de tentativas usadas.

3. Estruturas e Comandos Utilizados:

- `Random` para gerar número aleatório.
- `Console.WriteLine()` e `Console.ReadLine()` para entrada e saída.
- Conversão de string para inteiro (`int.Parse()`).
- Estrutura de repetição `do-while`.
- Estrutura condicional `if-else`.

4. Justificativa:

O uso do `do-while` é apropriado, pois a lógica exige pelo menos uma tentativa antes da verificação. A estrutura condicional permite dar feedback ao usuário, tornando o jogo interativo e educativo sobre controle de fluxo.

Código:

```
```csharp

using System;

class Program
{
 static void Main()
 {
 Random rnd = new Random();

 int numeroSecreto = rnd.Next(1, 101);

 int tentativa, tentativas = 0;

 Console.WriteLine("Adivinhe o número entre 1 e 100:");

 do
 {
 tentativa = int.Parse(Console.ReadLine());

 tentativas++;

 if (tentativa < numeroSecreto)
 Console.WriteLine("Mais alto!");
 else if (tentativa > numeroSecreto)
 Console.WriteLine("Mais baixo!");

 } while (tentativa != numeroSecreto);

 Console.WriteLine($"Parabéns! Você acertou em {tentativas} tentativas.");
 }
}
```

}

...