

## Exercícios de Fixação - Lógica de Programação

[Bóson Treinamentos em Tecnologia](#)

### ***Estruturas de Repetição***

Crie algoritmos para os itens a seguir, testando os programas no Portugol Studio.

#### **Exercício 1**

Crie um programa que mostre na tela o somatório total da sequência de 20 frações:  
 $1 + 1/2 + 1/3 + 1/4 + [...] + 1/20$ .

#### **Exercício 2**

Escreva um programa que mostre na tela todos os números pares entre 1000 e 1.

#### **Exercício 3**

Escreva um algoritmo que some os números de 1 a 10, mostrando na tela as somas parciais e, no final, o total.

A impressão deve seguir o padrão:

3 (soma do número 1 com o número 2)

6 (soma parcial com o número 3)

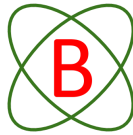
10 (soma parcial com o número 4)

[...]

55 (soma parcial com o número 10)

Soma total: 55

```
1
3
6
10
15
21
28
36
45
55
Soma total: 55
```



#### Exercício 4

Crie um jogo de adivinhação no qual o jogador #1 escolhe um número entre 1 e 10, e o jogador #2 insere números na tentativa de acertar o número escolhido pelo jogador 1.

Quando ele acertar, o algoritmo deverá informar que ele acertou o número  $x$  (escolhido pelo jogador #1) em  $y$  tentativas (quantidade de tentativas do jogador #2).

#### Exercício 5

Reescreva o jogo anterior para permitir que o jogador #1 insira apenas números entre 1 e 20, e o jogador #2 tenha apenas 4 tentativas para adivinhar o número.

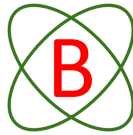
#### Exercício 6

Crie um programa para um sistema de votação com urna eletrônica. Neste sistema, os partidos válidos são os de números 43 e 51.

O eleitor (usuário) digita seu voto e pressiona a tecla *enter*, e logo após o programa pede para o próximo eleitor votar.

Caso não seja informado nenhum dos números de partidos válidos, o voto é considerado nulo, e caso seja informado o número zero, a votação é encerrada e então será exibido na tela o resultado da eleição, com o número de votos em cada partido, os votos nulos e a quantidade total de eleitores que votaram, como demonstra o esquema a seguir:

```
Resultados da eleição:  
Número total de votantes: 22  
Partido 43 teve 7 votos  
Partido 51 teve 12 votos  
Número de votos nulos: 3 votos
```



### Exercício 7

Elabore um algoritmo que pergunte a um usuário um número, calcule e mostre sua tabuada na tela, como no exemplo a seguir:

```
Digite um número para exibir sua tabuada:
6
1 x 6 = 6
2 x 6 = 12
3 x 6 = 18
4 x 6 = 24
5 x 6 = 30
6 x 6 = 36
7 x 6 = 42
8 x 6 = 48
9 x 6 = 54
10 x 6 = 60

Calcular outra tabuada?
Tecle s para "sim" ou outra tecla para encerrar:
```

### Exercício 8

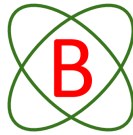
Crie um algoritmo para gerar conjuntos de números automaticamente. O programa deve perguntar ao usuário quantos conjuntos deseja gerar, e quantos números os conjuntos terão.

Por exemplo, 10 conjuntos com 3 números em cada, ou 5 conjuntos com 9 números em cada.

Mostrar na tela os valores, como mostra a ilustração a seguir:

```
Digite a quantidade de conjuntos a gerar: 6
Digite a quantidade de números por conjunto: 7

Conjunto: 1:    45    79    43    12    5    44    50
Conjunto: 2:    46    8    89    49    60    72    29
Conjunto: 3:    50    3    15    84    45    86    37
Conjunto: 4:    56    44    87    32    86    100    49
Conjunto: 5:    37    2    88    84    40    16    6
Conjunto: 6:    9    61    65    60    33    51    82
```



## Resolução dos Exercícios

### Exercício 1

```
programa
{
    const real numerador = 1.0
    inteiro denominador
    real soma = 0.0

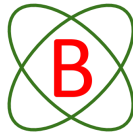
    funcao inicio()
    {
        para (denominador = 1; denominador <= 20; denominador++) {
            soma += numerador / denominador
        }

        escreva("Somatório das frações: " + soma)
    }
}
```

### Exercício 2

```
programa
{
    inteiro num = 1000

    funcao inicio()
    {
        enquanto(num > 0) {
            escreva(num + "\n")
            num -= 2
            // ou opcionalmente:
            // num = num - 2
        }
    }
}
```



### Exercício 3

```
programa
{
    inteiro num, soma = 0

    funcao inicio()
    {

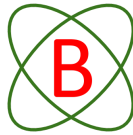
        para (num = 1; num <= 10; num++) {
            soma += num
            escreva(soma + "\n")
        }

        escreva("Soma total: " + soma)
    }
}
```

### Exercício 4

```
programa
{
    inteiro num, adivinha, tentativas = 0

    funcao inicio()
    {
        escreva("Jogador #1, digite um número entre 1 e 10: ")
        leia(num)
        limpa()
        enquanto (verdadeiro) {
            escreva("Jogador #2, qual é o número? ")
            leia(adivinha)
            tentativas++
            se (adivinha == num) {
                escreva("Você acertou em " + tentativas + " tentativas!")
                pare
            }
            escreva("Errou!\n")
        }
    }
}
```

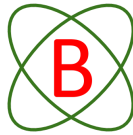


### Exercício 5

```
programa
{
    inteiro num, adivinha, tentativas = 0

    funcao inicio()
    {
        enquanto (verdadeiro) {
            escreva("Jogador #1, digite um número entre 1 e 20: ")
            leia(num)
            se ((num >= 1) e (num <= 20)) {
                pare
            }
            senao {
                escreva("Número fora do intervalo. Tente outra vez.\n")
            }
        }
        limpa()

        enquanto (tentativas < 4) {
            escreva("Jogador #2, qual é o número? ")
            leia(adivinha)
            tentativas++
            se (adivinha == num) {
                escreva("Você acertou em " + tentativas + " tentativas!")
                pare
            }
            escreva("Errou!\n")
        }
    }
}
```

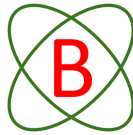


## Exercício 6

```
programa
{
    inteiro voto, votantes = 0
    inteiro voto43 = 0, voto51 = 0, votoNulo = 0

    funcao inicio()
    {
        enquanto (verdadeiro) {
            escreva("Candidatos: 43 e 51\n")
            escreva("0 encerra a votação\n")
            escreva("Digite seu voto: ")
            leia(voto)
            se (voto == 43) {
                voto43++
            }
            senao se (voto == 51){
                voto51++
            }
            senao se (voto == 0) {
                pare
            }
            senao {
                votoNulo++
            }
            votantes++
            limpa()
        }
        limpa()

        escreva("Resultados da eleição: \n")
        escreva("Número total de votantes: " + votantes + "\n")
        escreva("Partido 43 teve " + voto43 + " votos\n")
        escreva("Partido 51 teve " + voto51 + " votos\n")
        escreva("Número de votos nulos: " + votoNulo + " votos\n")
    }
}
```

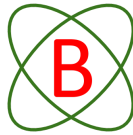


## Exercício 7

```
programa
{
    inteiro num, cont, x
    caracter continua = 's'

    funcao inicio()
    {
        enquanto (continua == 's') {
            escreva("Digite um número para exibir sua tabuada:\n")
            leia(num)
            para(cont = 1; cont <= 10; cont++) {
                escreva(cont + " x " + num + " = " + cont * num + "\n")
            }
            escreva("\nCalcular outra tabuada?\n")
            escreva("Tecla s para \"sim\" ou outra tecla para encerrar: ")
            leia(continua)
            limpa()
        }
    }
}
```





### Exercício 8

```
programa {  
    inclua biblioteca Util --> u  
  
    inteiro contA, contB, quantA, quantB  
  
    funcao inicio() {  
        escreva("Digite a quantidade de conjuntos a gerar: ")  
        leia(quantA)  
        escreva("Digite a quantidade de números por conjunto: ")  
        leia(quantB)  
        para(contA = 1; contA <= quantA; contA++) {  
            escreva("\nConjunto: " + contA + ":\t ")  
            para(contB = 1; contB <= quantB; contB++) {  
                inteiro n = u.sorteia(1,100)  
                escreva(n + "\t")  
            }  
        }  
    }  
}
```