

## Exercícios para iniciantes em Lógica de Programação usando o VisualG

### Declarações de variáveis:

- Inteiro → Variável do tipo inteiro.  
Exemplo: numeroInteiro : inteiro
- Real → Variável do tipo real, com casas decimais.  
Exemplo: numeroReal : real
- Caractere → Variável do tipo caractere.  
Exemplo: nome : caractere
- Lógico → Variável do tipo booleano.  
Exemplo: formado : logico

### Atribuições em variáveis e entrada de dados:

- Inteiro → Armazena valores numéricos inteiros.  
Exemplo: numeroInteiro := 7
- Real → Armazena valores numéricos reais, com casas decimais.  
Exemplo: numeroReal := 10.9977
- Caractere → Armazena cadeia de caracteres, Strings em outras linguagens.  
Exemplo: nome := "Sócrates"
- Lógico → Armazena valores booleanos, apenas *verdadeiro* ou *falso*.  
Exemplo: formado := verdadeiro ou formado := falso

### Entrada de dados via usuário - Comando LEIA:

- LEIA → Comando que recebe dados do usuário via teclado.  
Exemplo: leia(variavelOpcao)

### Impressão de dados na tela - Comandos ESCRIVA e ESCRIVAL:

- ESCRIVA → Comando que imprime valor na tela, sem pular linha.  
Exemplo: escreva(numeroInteiro)
- ESCRIVAL → Comando que imprime valor na tela, pulando linha ao final.  
Exemplo: escreval(numeroInteiro)
- Concatenando valores → Valor da variável + texto.  
Exemplo: escreval("Olá " + variavelNome + ", seja bem-vindo(a)!")

- Concatenando valores II → Valor da variável numérica + texto.  
Exemplo: escreval("O valor da variável é: ", numeroInteiro)

### Comentários no código:

- Comentário em linha → Comando para inserir comentários no código.  
Exemplo: // Esse é um comentário...  
*\*Em outras linguagens de programação existem os comentários em bloco, o qual não existe no VISUALG.*

### Operadores relacionais:

- Igual → Testa a igualdade entre valores.  
Exemplo: (a = b)
- Diferente → Testa a diferença entre valores.  
Exemplo: (a <> b)
- Maior → Testa se valor a é maior que valor b.  
Exemplo: (a > b)
- Menor → Testa se valor a é menor que valor b.  
Exemplo: (a < b)
- Maior ou Igual → Testa se valor a é maior ou igual ao valor b.  
Exemplo: (a >= b)
- Menor ou Igual → Testa se valor a é menor ou igual ao valor b.  
Exemplo: (a <= b)

### Operadores lógicos:

- E → Auxilia nas condicionais exigindo que os dois lados da expressão sejam verdadeiros.  
Exemplo: (nota > 7 e faltas < 10)
- OU → Auxilia nas condicionais exigindo que somente um dos lados da expressão seja verdadeiro.  
Exemplo: (nota > 7 ou faltas < 10)
- Não → Testa se o valor é *não* a.  
Exemplo: (não a)

### Condicionais:

- Se, Se Senão, Se Senão Se → Estruturas Condicionais.  
Exemplo:

## Algoritmo "Condicionais"

Var

// Seção de Declarações das variáveis

notaUm : real

notaDois : real

media : real

faltas : inteiro

Inicio

// Seção de Comandos, procedimento, funções, operadores, etc...

notaUm := 5

notaDois := 3

media := 0

faltas := 2

media := (notaUm + notaDois) / 2

// Média  $\geq 7$  e faltas  $\leq 10$

se (media  $\geq 7$ ) e (faltas  $\leq 10$ ) então

    escreval("Aprovado!")

fimse

// Média  $\geq 7$  e faltas  $\leq 10$  Aprovado

se (media  $\geq 7$ ) e (faltas  $\leq 10$ ) então

    escreval("Aprovado!")

senão

```
    escreval("Reprovado!")

fimse

// Média >= 7 e faltas <= 10 Aprovado

// Média >= 5 e faltas <= 10 Em recuperação.

se (media >= 7) e (faltas <= 10 ) então

    escreval("Aprovado!")

senão

    se (media >= 5) e (faltas <= 10 ) entao

        escreval("Em recuperação!")

    senão

        escreval("Reprovado!")

fimse

fimse
```

Finalgoritmo

### **Comandos de Repetição: Para, Enquanto e Repita.**

#### **Comandos de Repetição:**

- Para...faça → O exemplo abaixo imprime números de 1 a 5, mas pode usar outros valores.

Exemplo:

j : inteiro // Declarar variável controle do laço de repetição.

para j de 1 até 10 faça

escreva(j)

fimpara

- Enquanto...faça → O exemplo abaixo cria um laço que se repete até o usuário digitar 9.

Exemplo:

opcao : inteiro // Declarar variável controle do laço de repetição.

enquanto (opcao <> 9) faca

leia(opcao)

fimenquanto

- Repita...até → Idem ao *Enquanto*. A diferença é que o teste para saber se a variável tem o valor 9 é no final do laço.

Exemplo:

opcao : inteiro // Declarar variável controle do laço de repetição.

repita

leia(opcao)

ate (opcao = 9)

### Operadores Aritméticos:

- Soma → Soma de dois ou mais valores.  
Exemplo: total := num1 + num2
- Subtração → Subtração de dois ou mais valores.  
Exemplo: total := num1 - num2
- Divisão → Divisão de um valor.  
Exemplo: total := num1 / num2
- Multiplicação → Multiplicação de um ou mais valores.  
Exemplo: total := num1 \* num2
- Potenciação → Potenciação de um valor.  
Exemplo: total := 5 ^ 3 *O resultado será 125.*

- Mod → Resto de uma divisão entre dois valores inteiros.  
Exemplo: total := 5 mod 2 *O resultado será 1.*
- Quociente → Retorna a parte inteira de uma divisão entre dois valores inteiros.  
Exemplo: total := 100 \ 3 *O resultado será 33.*

•

### **Comando de Seleção - Escolha.**

#### **Comando de Seleção:**

- Escolha → O comando escolha permite várias opções e uma "default", caso não se enquadre nas demais.

Exemplo:

Algoritmo "Escolha"

Var

// Seção de Declarações das variáveis

letra : caractere

Início

escreval("Escolha uma letra: ")

leia(letra)

escolha letra

caso "A"

escreval("Letra A")

caso "B"

escreval("Letra B")

outrocaso

escreval("Nem A, nem B, Outra letra")

fimescolha

Fimalgoritmo

### **Declarações de vetores:**

- Inteiros → Vetor com valores Inteiros.  
Exemplo: vetorDeInteiros: vetor [1..10] de inteiro
- Reais → Vetor com valores Reais.  
Exemplo: vetorDeReais: vetor [1..10] de real
- Caracteres → Vetor com caracteres.  
Exemplo: vetorDeCaracteres: vetor [1..10] de caractere

### **Atribuições em vetores:**

- Inteiro → Atribuição de valor inteiro na posição 1 do vetor.  
Exemplo: vetorDeInteiros[1] := 3558
- Real → Atribuição de valor real na posição 4 do vetor.  
Exemplo: vetorDeReais[4] := 3.14
- Caractere → Atribuição de valor em caracteres na posição 2 do vetor.  
Exemplo: vetorDeCaracteres[2] := "Sócrates"

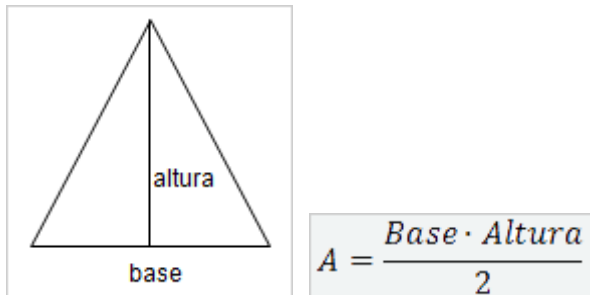
### **Imprimindo valores armazenados em vetores:**

- Inteiro → Imprime valor inteiro armazenado na posição 1 do vetor.  
Exemplo: escreva(vetorDeInteiros[1])
- Real → Imprime valor real armazenado na posição 1 do vetor.  
Exemplo: escreva(vetorDeReais[1]:2:5) // Note que é acrescentado :2:5 após o vetor. Esse valor é a quantidade de casas decimais de cada lado do número que desejamos imprimir.
- Caractere → Imprime valor em caracteres (literal, string) armazenado na posição 1 do vetor.  
Exemplo: escreva(vetorDeCaracteres[1])

### **Desafios:**

1) Este exercício é apenas um desafio de lógica. Escreva um algoritmo que armazene o valor 10 em uma variável A e o valor 20 em uma variável B. A seguir (utilizando apenas atribuições entre variáveis) troque os seus conteúdos fazendo com que o valor que está em A passe para B e vice-versa. Ao final, escrever os valores que ficaram armazenados nas variáveis.

2) Escreva um algoritmo para ler as dimensões de um triângulo (base e altura), calcular e escrever a área do triângulo. Sabendo que para calcular a área devemos usar a fórmula a seguir;



3) Faça um algoritmo que leia a idade de uma pessoa expressa em anos, meses e dias e escreva a idade dessa pessoa expressa apenas em dias. Considerar ano com 365 dias e mês com 30 dias.

### Soluções dos algoritmos.

#### Exercício 1

##### Algoritmo "Troca de Valores"

Var

a, b, intermediario: inteiro // crie duas variáveis do tipo inteiro.

Inicio

// Seção de Comandos, procedimento, funções, operadores, etc...

Escreva("Olá, digite um valor para A:")

leia(a)

Escreva("Olá de novo, digite agora outro valor para B:")

leia(b)

// Exibe para o usuário os valores digitados.

Escreva("Muito bem! Você digitou os valores para A:", a, " e B:", b)

// Fazemos a troca dos valores das variáveis,

// precisamos de uma variável intermediária para fazer a troca

intermediario <- a

a <- b

b <- intermediario

Escreva("Se invertermos os valores de A e B teremos A:", a, " e B:", b)



## Fimalgoritmo

### Exercício 2

Algoritmo "Triangulo Calculator 3100"

Var

base, altura, area: real // crie duas variáveis do tipo inteiro.

Inicio

// Seção de Comandos, procedimento, funções, operadores, etc...

Escreval("\*\*\*\*\* SUPER TRIANGULO CALCULATOR 3100 \*\*\*\*\* ")

Escreval()

Escreval()

Escreva("Olá, informe a base do triângulo:")

leia(base)

Escreva("Olá, informe a altura do triângulo:")

leia(altura)

area <- (base \* altura)/2

Escreva("Esse triangulo tem uma área de ", area)

## Fimalgoritmo

### Exercício 3

Algoritmo "IDADE CALCULATOR 5600"

Var

diaNascimento, mesNascimento, anoNascimento: inteiro

calculaAno, calculaMes, calculaDias: inteiro

Inicio

// Seção de Comandos, procedimento, funções, operadores, etc...

Escreval("\*\*\*\*\* SUPER IDADE CALCULATOR 5600 \*\*\*\*\* ")

Escreval()

Escreval()

```
Escreval("Digite o dia em que nasceu: ")  
leia(diaNascimento)
```

```
Escreval("Digite o mes em que nasceu (digite o mês em número, de 1 à 12): ")  
leia(mesNascimento)
```

```
Escreval("Digite o ano em que nasceu: ")  
leia(anoNascimento)
```

```
Escreval("***** ")
```

```
calculaAno <- (2017 - anoNascimento) * 365  
calculaMes <- (mesNascimento * 30) - (30 - diaNascimento)  
calculaDias <- CalculaAno + calculaMes
```

```
Escreva("Você possui", calculaDias, " dias de vida")  
Escreval()  
Escreval()
```

Fimalgoritmo