

O que é lógica de programação?

Lógica de programação é como dar instruções para um robô. Imagine que você tem um robô amigável que faz coisas legais quando você pede a ele. Para fazer o robô funcionar, você precisa dar a ele um conjunto de instruções claras e simples, uma de cada vez.

Por exemplo, pense em uma receita para fazer um bolo de chocolate. A receita é como um programa de computador.

Ela diz o que fazer em uma ordem específica. Primeiro, você mistura a farinha e o açúcar, depois adiciona o chocolate, os ovos e o leite. Finalmente, você coloca a massa no forno. Se você mudar a ordem das etapas, o bolo não sairá direito!

Espero que saiba!

Operadores matemáticos:

1. **Adição (+):** Usado para adicionar dois valores. Por exemplo, $a + b$ adiciona os valores de a e b .
2. **Subtração (-):** Usado para subtrair o valor à direita do valor à esquerda. Por exemplo, $a - b$ subtrai b de a .
3. **Multiplicação (*):** Usado para multiplicar dois valores. Por exemplo, $a * b$ multiplica os valores de a e b .
4. **Divisão (/):** Usado para dividir o valor à esquerda pelo valor à direita. Por exemplo, a / b divide a por b . É importante evitar a divisão por zero, pois isso resultaria em erro.
5. **Resto (%):** Usado para encontrar o resto da divisão do valor à esquerda pelo valor à direita. Por exemplo, $a \% b$ retorna o resto da divisão de a por b .
6. **Potenciação (^):** Usado para elevar um número à potência de outro número. Por exemplo, $a ^ b$ eleva a à potência de b .
7. **Radiciação (raiz quadrada):** No VisuAlg, a raiz quadrada de um número pode ser encontrada usando a função **sqrt(valor)**. Por exemplo, **sqrt(a)** retorna a raiz quadrada de a .

Variáveis:

Variáveis são espaços de armazenamento na memória de um computador que são usados para armazenar dados temporários ou valores manipulados por um programa de computador. Elas são como “caixas” ou “contêineres” que guardam informações que podem ser acessadas e modificadas durante a execução de um programa.

Quando você escreve um programa de computador, você usa variáveis para armazenar dados como números, textos, caracteres, valores lógicos (verdadeiro ou falso) e outros tipos de informações. Cada variável tem um nome único que a identifica e um tipo que determina o tipo de dados que ela pode armazenar (inteiro, real, caractere, etc.). Neste começo focaremos apenas nas variáveis do tipo inteiro e real.

Inteiro (Int): Variáveis inteiras são usadas para armazenar números inteiros, ou seja, números sem casas decimais. Por exemplo:

```
var
idade: inteiro
```

Real: Variáveis reais são usadas para armazenar números com casas decimais. Por exemplo:

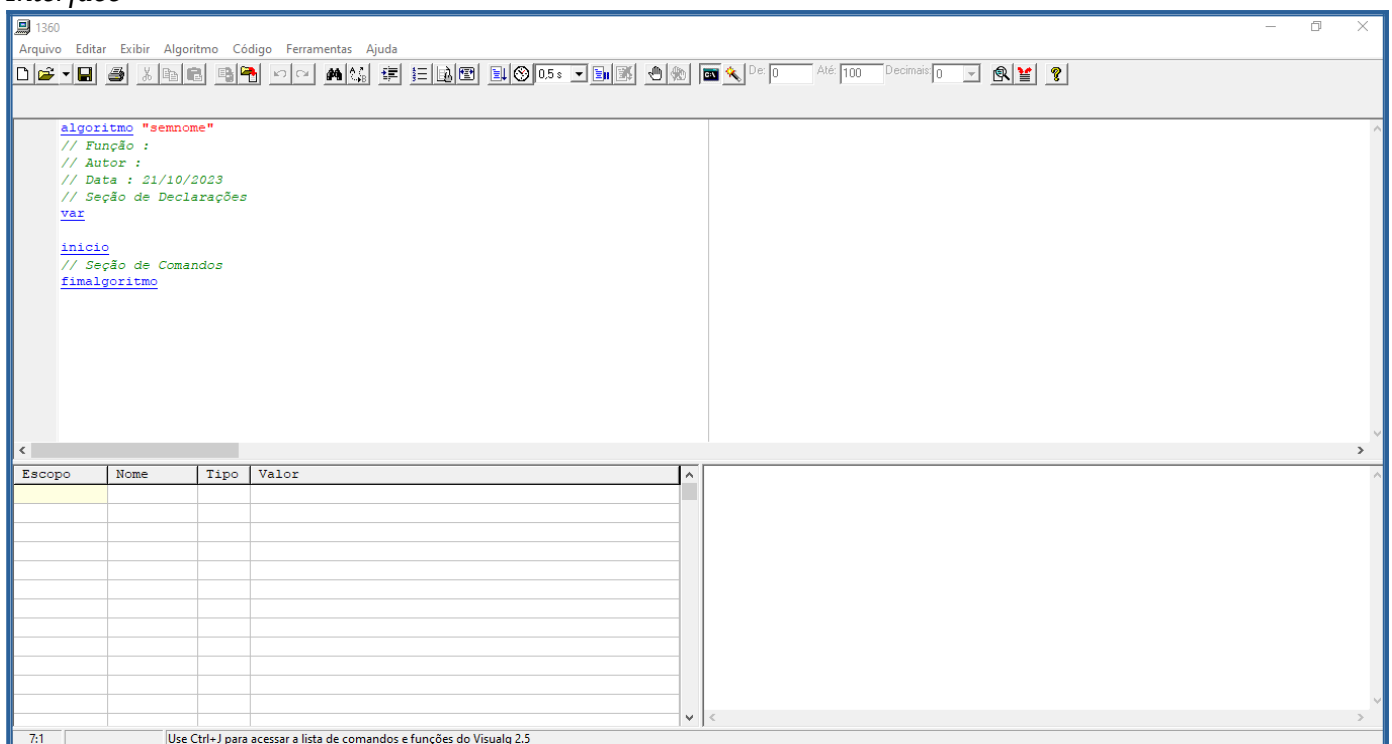
```
var
preco: real
```

O que é necessário?

O **VisuAlg** é um ambiente de desenvolvimento integrado (IDE) muito utilizado por quem está aprendendo a programar em linguagem de pseudocódigo. Pseudocódigo é uma forma de escrever algoritmos de programação usando linguagem humana, sem a necessidade de aprender uma linguagem de programação complexa.

Você pode baixar [aqui!](#)

Interface



Exemplo de código utilizando o VisuAlg

```
algoritmo "semnome"
// Função :
// Autor :
// Data : 21/10/2023
// Seção de Declarações
var
numero1, numero2, resultado_final: real
inicio
// Seção de Comandos
escreval("----PROGRAMA PARA SOMAR DOIS NÚMEROS----")
escreva("Digite o primeiro número: ")
leia(numero1)
escreva("Digite o segundo número: ")
leia(numero2)
resultado_final <- numero1 + numero2
escreva("O resultado final será:", resultado_final)
fimalgoritmo
```

Vamos entender o que está nesta imagem.

Primeiramente temos a seção var onde nós colocamos nossas variáveis, para somar dois números criamos duas variáveis chamadas **numero1**, **numero2** e uma variável chamada **resultado_final** que receberá os dados de **numero1** e **numero2** e irá somá-las. Não use caracteres especiais, espaço e etc para nomear elas, use **snake_case** que separa as palavras assim_por_exemplo ou **camelCase** que ficaria assim porExemplo.

Após declararmos as variáveis, podemos começar a escrever nosso código na seção inicio onde temos **escreval** e **escreva**, ambas fazem a mesma coisa que é mostrar um texto na tela, **escreval** apenas cria uma quebra de linha, fazendo com que o próximo texto apareça embaixo, ao invés do lado. Todo texto deve estar dentro de aspas.

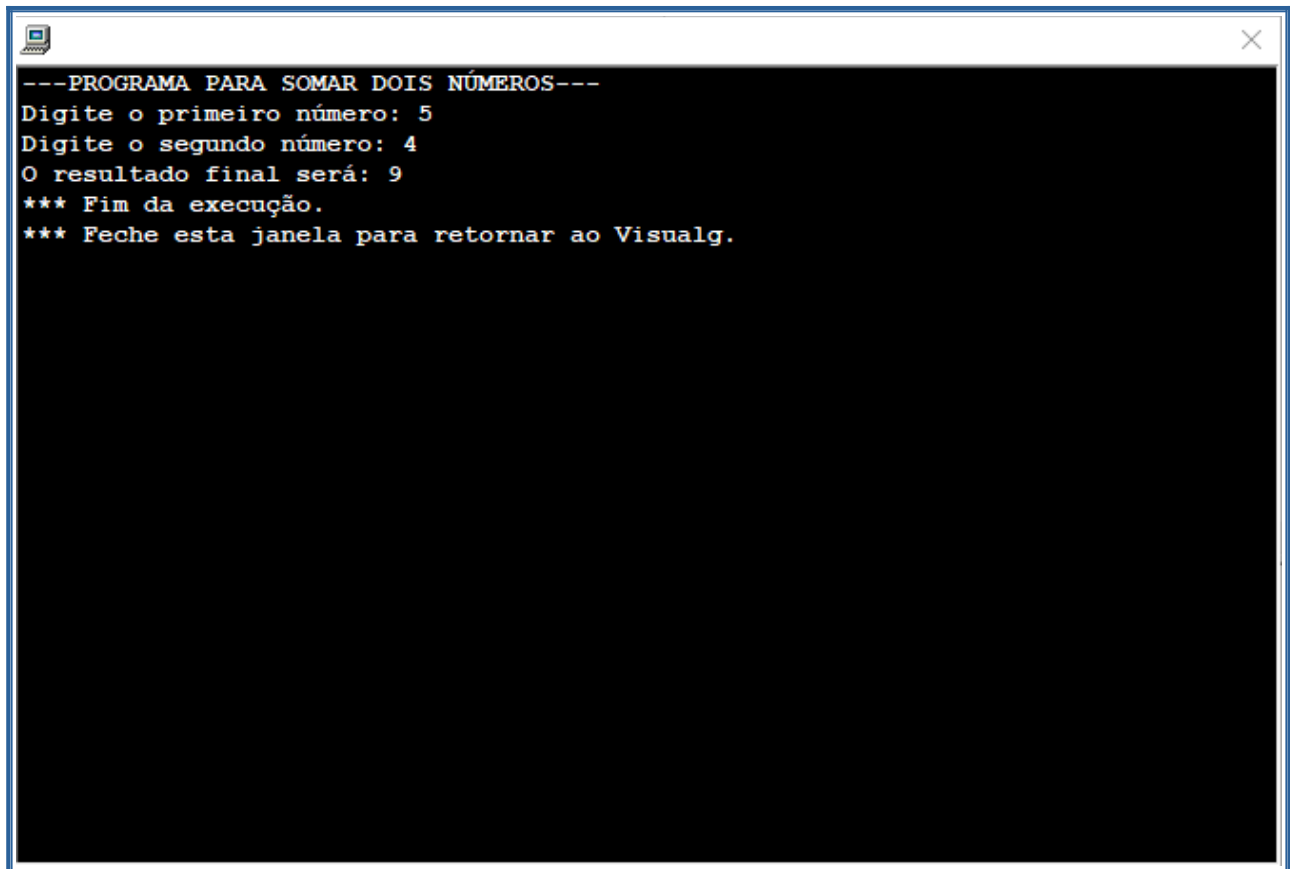
O comando **leia** receberá o que o usuário escreveu e atribuirá à variável que criamos, no caso **numero1** e assim no caso da variável **numero2**.

Recebemos os dados das duas variáveis e queremos somá-las, utilizaremos a variável **resultado_final** para receber os dados das duas, o sinal de atribuição é o <- (menor que e o traço de menos juntos). Só precisamos citar cada uma e somar com o sinal de + pois o valor que digitamos já está armazenado em cada uma.

Por fim, nós queremos que o resultado seja mostrado e nisso utilizaremos também o comando **escreva**, mas perceba que temos dois conteúdos dentro dela, um texto e a variável, como sabemos o texto deve ficar entre aspas e para citar a variável nós colocamos uma **vírgula** para separar cada um, mas sempre dentro dos parênteses.

No final temos o comando fimalgoritmo.

Ao rodarmos teremos a seguinte imagem:

A screenshot of a Visual Basic console window. The window has a title bar with a small icon on the left and a close button (X) on the right. The background is black, and the text is displayed in a light blue or cyan monospaced font. The text content is as follows:

```
---PROGRAMA PARA SOMAR DOIS NÚMEROS---  
Digite o primeiro número: 5  
Digite o segundo número: 4  
O resultado final será: 9  
*** Fim da execução.  
*** Feche esta janela para retornar ao Visualg.
```

Alguns comandos úteis são o F9 que irá rodar o seu código e ESC para fechar a janela após ele rodar.

Exercícios

Busque replicar o código. Utilize outros sinais aritméticos para explorar diferentes resultados.

Desafios

1. A imobiliária Imóbilis vende apenas terrenos retangulares. Faça um algoritmo para ler as dimensões de um terreno e depois exibir a área do terreno.
2. Faça um algoritmo para calcular quantas ferraduras são necessárias para equipar todos os cavalos comprados para um haras.

Gabarito

1 -

```
algoritmo "semnome"  
// Função :  
// Autor :  
// Data : 14/10/2023  
// Seção de Declarações  
var  
lado, comprimento, area: real  
inicio  
// Seção de Comandos  
escreva("Digite o valor do lado do retângulo: ")  
leia(lado)  
escreva("Digite o valor do comprimento do retângulo: ")  
leia(comprimento)  
area <- lado*comprimento  
escreva("A área do terreno é ", area,"m")  
finalgoritmo
```

2 -

```
algoritmo "semnome"  
// Função :  
// Autor :  
// Data : 14/10/2023  
// Seção de Declarações  
var  
ferraduras, cavalos: inteiro  
inicio  
// Seção de Comandos  
escreva("Quantos cavalos serão comprados? ")  
leia(cavalos)  
ferraduras <- cavalos*4  
escreva("Será necessário comprar ", ferraduras," ferraduras")  
finalgoritmo
```

Referências

Lista de Exercícios 1 - Algoritmos Sequenciais — Jonathan Pereira. Disponível em:
<<https://docentes.ifrn.edu.br/jonathanpereira/disciplinas/algoritmos/lista-de-exercicios-1/view>>.
Acesso em: 21 out. 2023.