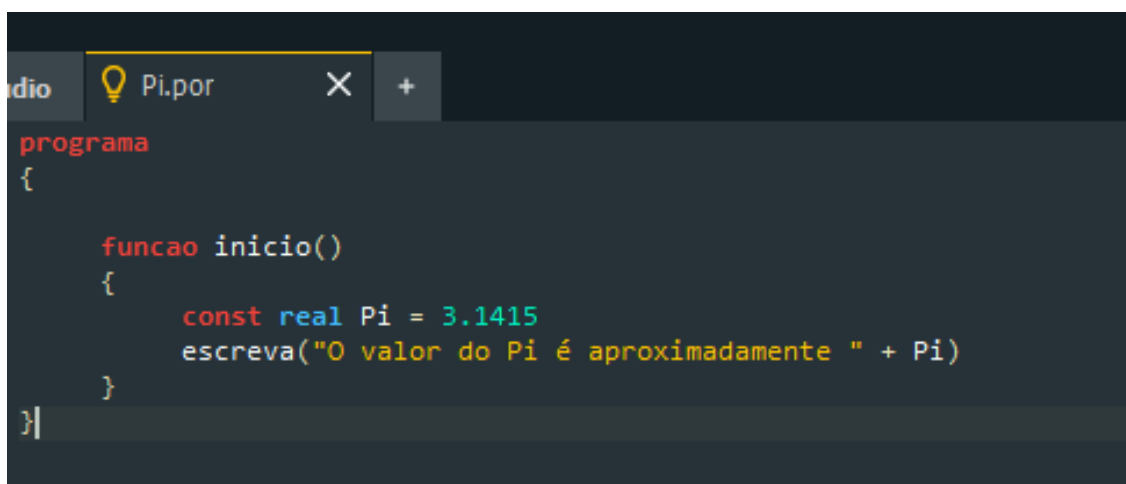


Constantes

Uma constante segue as mesmas regras de variável, mas com certeza de que seu valor não será alterado durante a execução do programa, ou seja, será sempre o mesmo, sendo obrigatória a atribuição de um valor no momento da declaração.

Um exemplo de uma constante matemática é o número Pi, que é um valor fixo de aproximadamente 3,1415 e continuará assim até o final da execução.





Exemplo:



```
programa
{
    funcao inicio()
    {
        const real Pi = 3.1415
        escreva("O valor do Pi é aproximadamente " + Pi)
    }
}
```

Operadores

Os operadores são utilizados para representar expressões de cálculo, comparação, condição e atribuição. Para construção de algoritmos temos os seguintes operadores:

-  Atribuição
-  Aritméticos
-  Relacionais
-  Lógicos

Operadores de atribuição

Um dos operadores mais utilizados na programação é o operador de atribuição, representado pelo sinal de =.

Exemplos em Portugol Studio

```
nomeCliente = "João da Silva"
resultado = a + 5
```

valor = 3.5

Operadores aritméticos

Chamamos de operadores aritméticos o conjunto de símbolos que representam as operações básicas de matemática.

Operação	Símbolo	Prioridade
Adição	+	1
Subtração	-	1
Multiplicação	*	2
Divisão	/	2
Resto da divisão inteira	%	2

Operadores relacionais

Operadores relacionais são aqueles que comparam dois valores (variáveis, constantes ou chamadas de funções) e/ou expressões e verificam quem é maior ou menor e/ou se existe igualdade entre eles. O resultado dessa comparação é sempre um valor lógico (booleano) verdadeiro ou falso)

Operação	Símbolo
Maior	>
Menor	<
Maior ou igual	>=
Menor ou igual	<=
Igual	==
Diferente	!=

Operação	Resultado
3 > 4	Falso
7 != 7	Falso
9 == 10 - 1	Verdadeiro
33 <= 100	Verdadeiro
6 >= 5 + 1	Verdadeiro

Operadores lógicos

Operadores são usados para concatenar ou associar expressões que estabelecem uma relação de comparação entre valores. Os resultado dessas expressões é sempre um valor lógico, verdadeiro ou falso, uma vez que operam sobre valores booleanos

Operador	Prioridade
ou	1
e	2
nao	3

Operações	Resultado
$5 > 3$ e $2 < 1$	falso
nao ($8 < 4$)	verdadeiro
$1 > 3$ ou $1 \leq 1$	verdadeiro

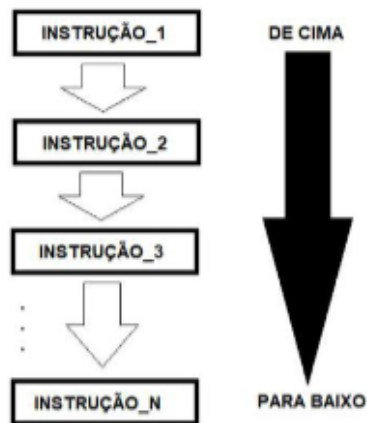
Estrutura Sequencial e de Seleção

Construir um algoritmo pressupõe obedecer uma ordem predeterminada a fim de que sejam dados os passos corretos, à luz de certas regras dentro de uma sequência lógica, para resolver o problema computacional. A seguir, aprenderemos as estruturas usadas na construção de algoritmos, com alguns exemplos em fluxograma e pseudocódigo.

Estrutura Sequencial

É uma estrutura que realiza um conjunto predeterminado de instruções de forma seqüencial, de cima para baixo, na ordem em que foram declaradas.

Figura 1 – Sintaxe da estrutura sequencial



A estrutura seqüencial de um algoritmo pode ser facilmente representada por meio de um fluxograma e pseudocódigo, conforme os seguintes exemplos.

Exemplo1: Considere um algoritmo que somará dois números inteiros quaisquer:

Fluxograma



Pseudocódigo

```

programa
{
    funcao inicio()
    {
        real numero1
        real numero2
        real resultado

        escreva("Digite a numero1:")
        leia (numero1)
        escreva("Digite o numero2:")
        leia (numero2)
        resultado = numero1 + numero2
        escreva(resultado)
    }
}

```

Estrutura de Seleção

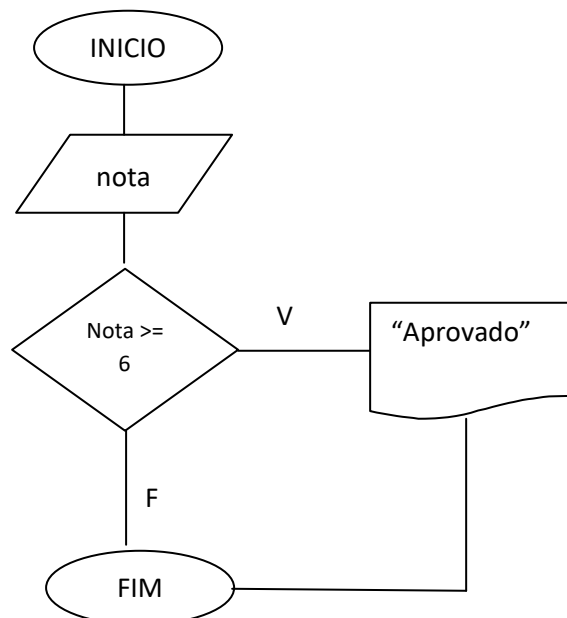
Uma estrutura de seleção viabiliza escolher um grupo de ações (bloco) a ser executado quando determinadas condições, representadas por expressões lógicas ou relacionais, são satisfeitas ou não. Os tipos de seleção são:

- 🖥️ Simples
- 🖥️ Composta
- 🖥️ Encadeada
- 🖥️ Múltipla escolha

Estrutura de Seleção Simples (se-então)

As estruturas de seleção simples são utilizadas para testar certa condição antes de executar determinada ação. Se a condição for atendida, um conjunto de instruções deverá ser executada; se não for atendida, o fluxo de execução do algoritmo seguirá após o fim do bloco de decisão.

Fluxograma



Pseudocódigo

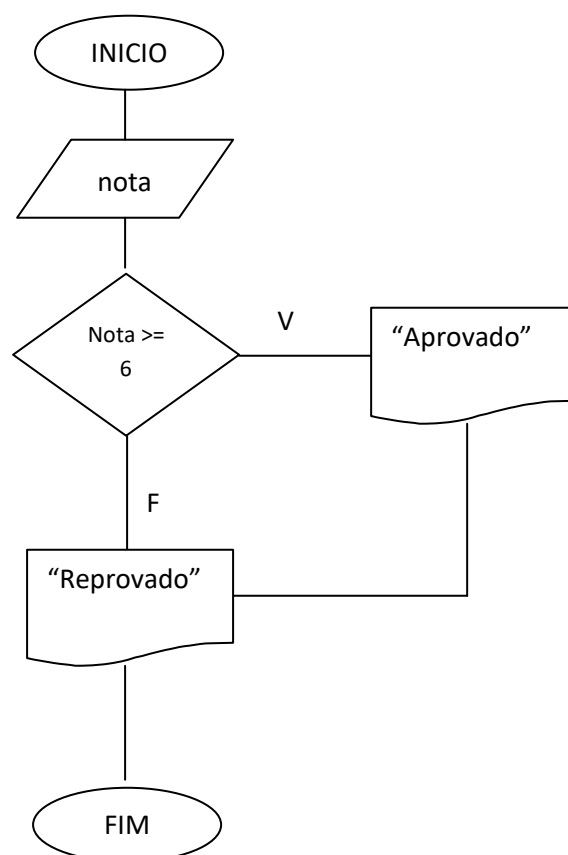
```
programa
{
    funcao inicio()
    {
        real nota = 8.0

        se (nota >= 6)
            escreva("Aprovado")
    }
}
```

Estrutura de seleção composta (se-então-senão)

A estrutura composta prevê dois conjuntos de instruções: um conjunto caso a resposta da condição resulte verdadeira; o outro, por sua vez, se resultar falsa.

Fluxograma



Pseudocódigo

```
programa
{
    funcao inicio()
    {
        real nota = 2.0

        se (nota >=6)
            escreva("Aprovado")
        senao
            escreva("Reprovado")
    }
}
```

Referência Bibliográfica

ARAUJO, Sandro: Lógica de Programação e Algoritmos. Curitiba: Contentus, 2020