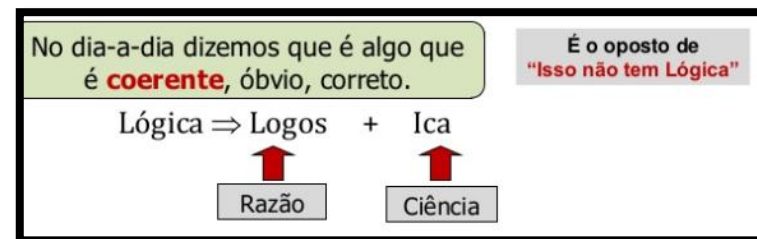


Lógica de Programação

Conceitos Iniciais: Lógica e Algoritmos e VisualG

Lógica



→ Consiste em utilizar de princípios e conhecimentos a fim de buscar atingir um raciocínio correto.

→ É necessário para formular uma **sequência de raciocínio** que permita a você utilizar de premissas e informações para ser possível obter uma **conclusão**.

Raciocínio Lógico

- Evidências/Premissas/Informações + Análise = Conclusão
- Evidências/Premissas/Informações **erradas** + Análise = Conclusão **falha**
- Evidências/Premissas/Informações + Análise = Conclusão **errada** **falha**
- Evidências/Premissas/Informações + Análise = Conclusão **falha**



Exemplos dia a dia:

- Cálculo matemático?
- Como fazer uma receita?
- Um processo comum do dia a dia
- Um processo técnico, profissional?
- Como chegar em um lugar?
- Como jogar um jogo?

Todos os peixes vivem na água. (premissa1)
Golfinhos são peixes. (premissa 2)
 \therefore Golfinhos vivem na água. (conclusão)

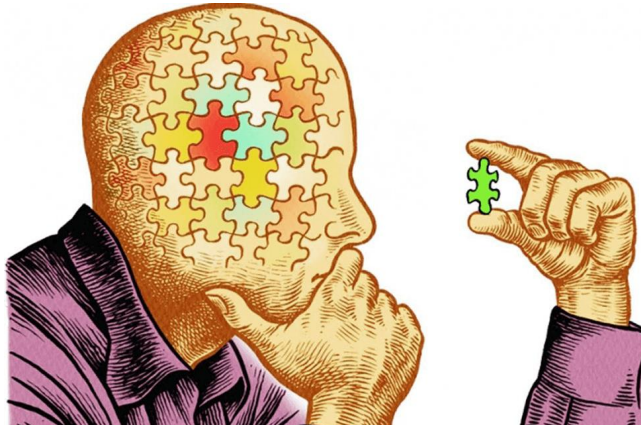


As coisas mais simples podem ser descritas por uma sequência lógica
Ex: "Chupar uma bala"

Lógica

Todo homem é mortal.
Sócrates é homem.
Logo, Sócrates é mortal.

Se $A = B$
Se $B = C$
Logo, $A = C$



$$2X + Y = 10$$

$$X + Y = 7$$

$X = 2$ e $Y = 3$ (V ou F?)

$X = 3$ e $Y = 4$ (V ou F?)

Qual é o valor da sequencia lógica?

1, 3, 5, 7, ?

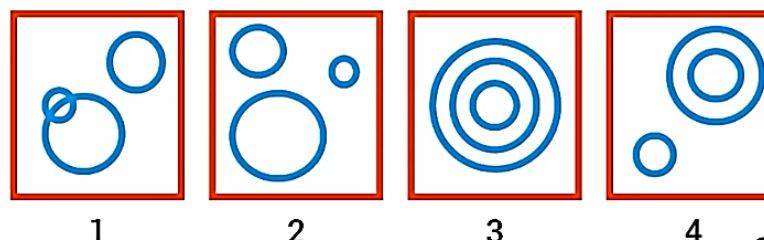
1, 1, 2, 3, 5, 8, ?

S, T, Q, ?, S, S, D

AC BD CE DF ?

SEQUÊNCIAS ENVOLVENDO FIGURAS

EXEMPLO 02



Animais, gatos, canetas

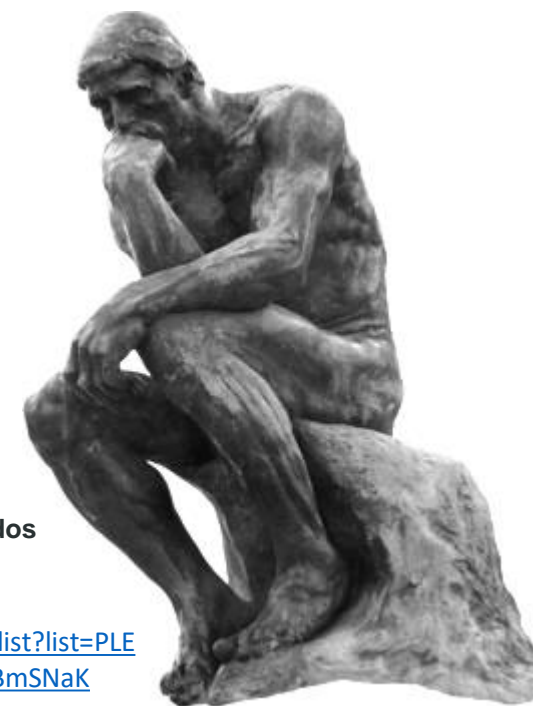
- a) 1
- b) 2
- c) 3
- d) 4

Seja Trainee : Testes de raciocínio lógico | Ebook e Exercícios resolvidos

<https://sejatraineer.com.br/testes-de-raciocinio-logico-dos-processos-traineer/>

Noções de Lógica (playlist)

<https://www.youtube.com/playlist?list=PLEfwqyY2ox85ShbZ1O0fXmkUxavBmSNak>



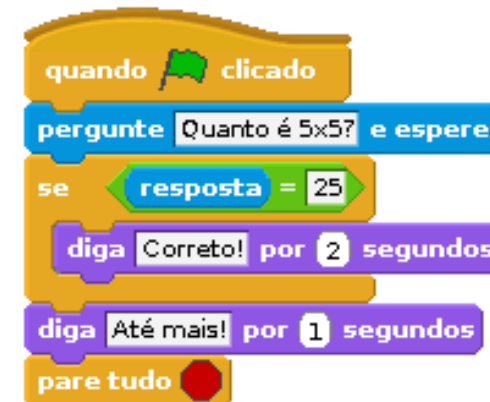
Link de um Drive com material sobre lógica para se preparar em testes e concursos

https://drive.google.com/drive/folders/1EuWOLylLZtkMvR08neVanF--4Ycp_4QB?usp=sharing

A Lógica de Programação e Algoritmos

→ Uma técnica de encadear pensamento para atingir um determinado **objetivo**.

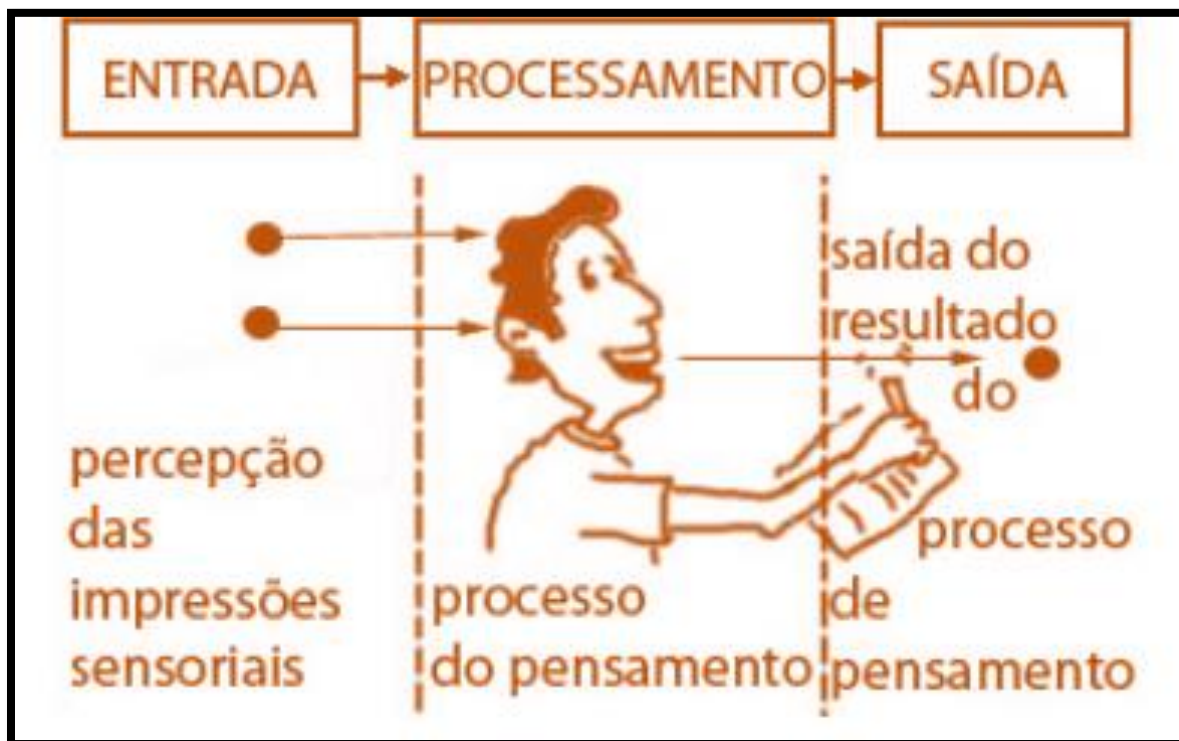
→ sequência lógica = que é um conjunto de passos a serem executados → **ALGORITMO**



<https://scratch.mit.edu/about>

Um ALGORITMO funcional deve

1. Ter um número **finito** de passos;
2. Que os passos sejam **bem definidos**;
3. Que existam **zero ou mais entradas**, tomadas em conjunto e bem estruturadas;
4. Existam **uma ou mais saídas**;
5. Que exista uma **condição de fim** sempre bem definida assim que o algoritmo completar sua tarefa para qualquer entrada num tempo determinado.
- 6- As frases (instruções) de um algoritmos geralmente são compostas por **verbos** no **IMPERATIVO** ou **INFINITIVO**



As 3 partes de um algoritmo

A Lógica de Programação e Algoritmos

Problema

Fazer um bolo de chocolate.

Saída

Qual o **resultado** esperado?

Passos

O que é preciso para se obter o resultado desejado?

☐ **Ingredientes**

Passos

Qual a sequência necessária (de uso dos **ingredientes**) para se obter o resultado desejado?



Dados de Entrada

Ingredientes:

- ☐ 5 ovos.
- ☐ 250 gramas de margarina cremosa.
- ☐ 2 xícaras (chá) de açúcar.
- ☐ 1 xícara (chá) de farinha de trigo.
- ☐ 1 xícara (chá) de chocolate.
- ☐ 200 gramas de côco ralado.
- ☐ 1 copo de leite.
- ☐ 1 colher (sopa) de fermento.

Processamento

Modo de Preparo

- ☐ Bata a margarina, as gemas e o açúcar até ficar cremoso.
- ☐ Junte o leite, o côco e a farinha e continue batendo.
- ☐ Acrescente o fermento e, por último, as claras em neve.
- ☐ Unte uma forma com manteiga e leve ao forno para assar.

A Lógica de Programação e Algoritmos

Escreva um algoritmo para ler dois números (a e b) e trocar os seus valores. Exibir os valores de a e b após a troca



- **SAÍDA:** Quais os resultados que o algoritmo deve fornecer ?
- **ENTRADA:** Quais os insumos necessários para se obter a saída?
- **PROCESSAMENTO:** Como transformar os insumos na saída?

- **SAÍDA:** valores de a e b (trocados)
- **ENTRADA:** dois números (A, B)
- **PROCESSAMENTO:**
 - $A = B$
 - $B = A$

```
5 inteiro a, b
6 escreva ("Digite o valor de a: ")
7 leia(a)
8 escreva ("Digite o valor de b: ")
9 leia(b)
10 a = b
11 b = a
12 escreva("\n a = ", a)
13 escreva("\n b = ", b)
```

Declaração de Variáveis

Entrada de Dados

Processamento

Saída de Dados

OBS: Esse programa não atende ao enunciado de forma correta, possui um **erro!**
(Trabalharemos na correção em "Teste de Mesa")

Representação de Algoritmos

ALGORITMO = Sequência lógica

Exemplo 1

DESCRIÇÃO NARRATIVA

diretamente em linguagem natural



Algoritmo Exemplo: FRITAR UM OVO

1. Pegue a frigideira, ovo, óleo e sal
2. Coloque o óleo na frigideira
3. Acenda o fogo
4. Coloque a frigideira no fogo
5. Espere o óleo esquentar
6. Quebre o ovo
7. Despeje o ovo no óleo quente
8. Coloque o sal
9. Retire quando estiver pronto
10. Desligue o fogo

Representação de Algoritmos

ALGORITMO = Sequência lógica

DESCRIÇÃO NARRATIVA diretamente em linguagem natural

Exemplo 2

Obter as quatro notas do aluno;

Somar as quatro notas obtidas;

Dividir o resultado da adição por quatro;

Se a média obtida for maior ou igual a 7 (sete):

O aluno foi aprovado.

Caso contrário:

O aluno foi reprovado.



$$\text{Média} = \frac{\text{nota1} + \text{nota2} + \text{nota3} + \text{nota4}}{4}$$

Quais serão as **saída de dados**?
Quais são os **dados de entrada**?
Qual será o **processamento** a ser utilizado?

Exemplo 3 – Trocar uma lâmpada

- Verificar se a lâmpada está quebrada;
- Comprar outra lâmpada;
- Pegar a escada;
- Verificar se a energia está desligada;
- Subir na escada;
- Afrouxar a lâmpada e retirar do bocal;
- Descer da escada e pegar outra lâmpada;

- Subir na escada;
- Rosquear a outra lâmpada no bocal;
- Descer da escada;
- Ir ao interruptor e verificar se a lâmpada acende;
- Retirar a escada do lugar e guardar;
- Descartar a lâmpada ruim.



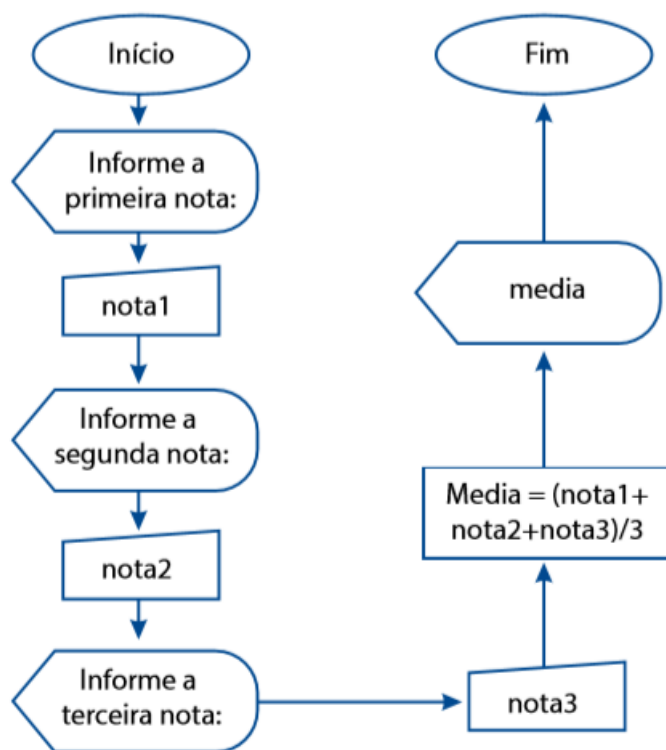
É pouco utilizada na prática, pois esse tipo de representação quando não é bem estruturada dá margem a **ambiguidades e imprecisões**.

Exemplo: “pegar a escada”, nesse caso precisaria descrever como vou pegar a escada, e ainda mais, descobrir primeiro onde ela está. Só assim seriam diminuídas as chances de se cometer algum tipo de equívoco.

Representação de Algoritmos

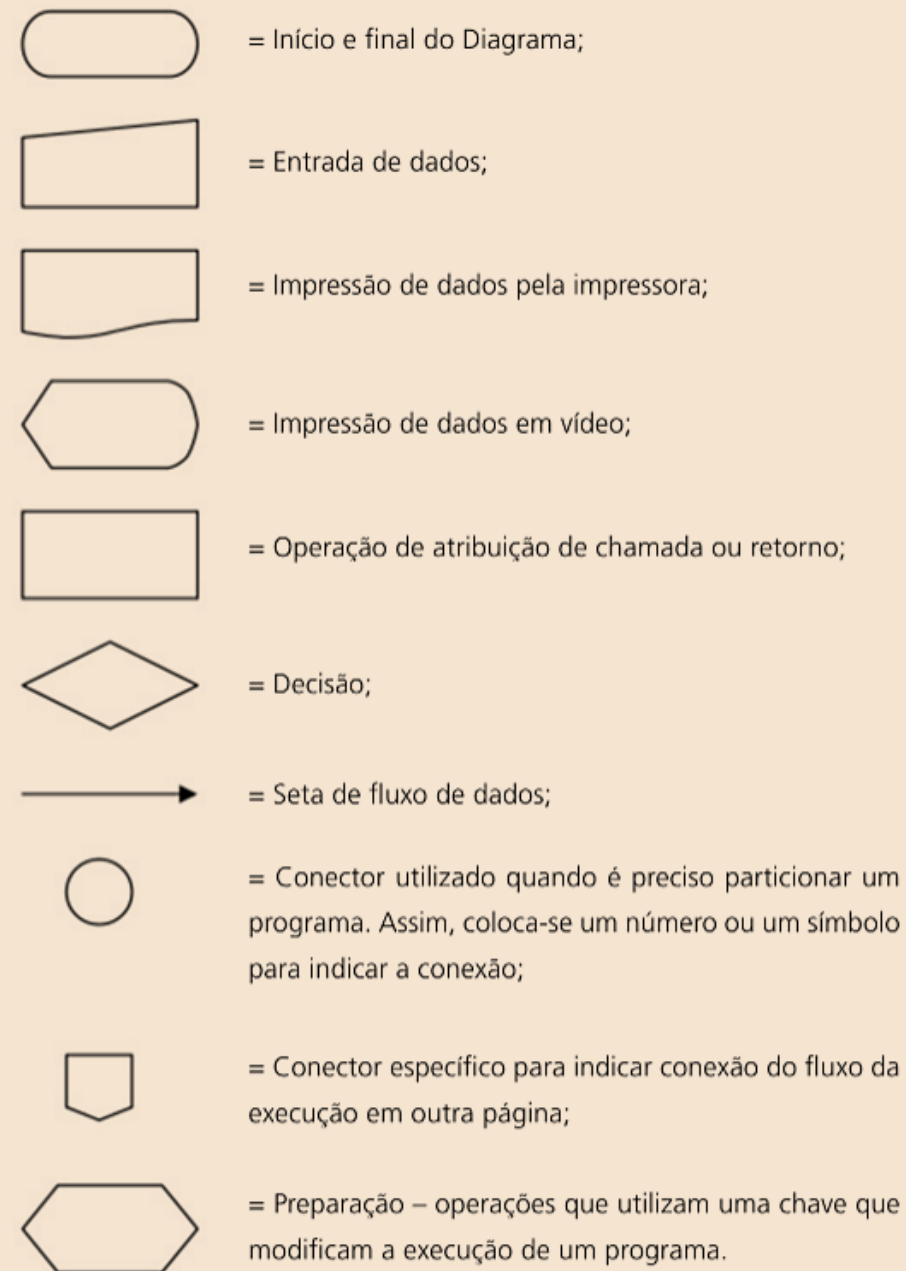
ALGORITMO = Sequência lógica

FLUXOGRAMA OU DIAGRAMA DE BLOCOS



Quais serão as **saída de dados**?
Quais são os **dados de entrada**?
Qual será o **processamento** a ser utilizado?

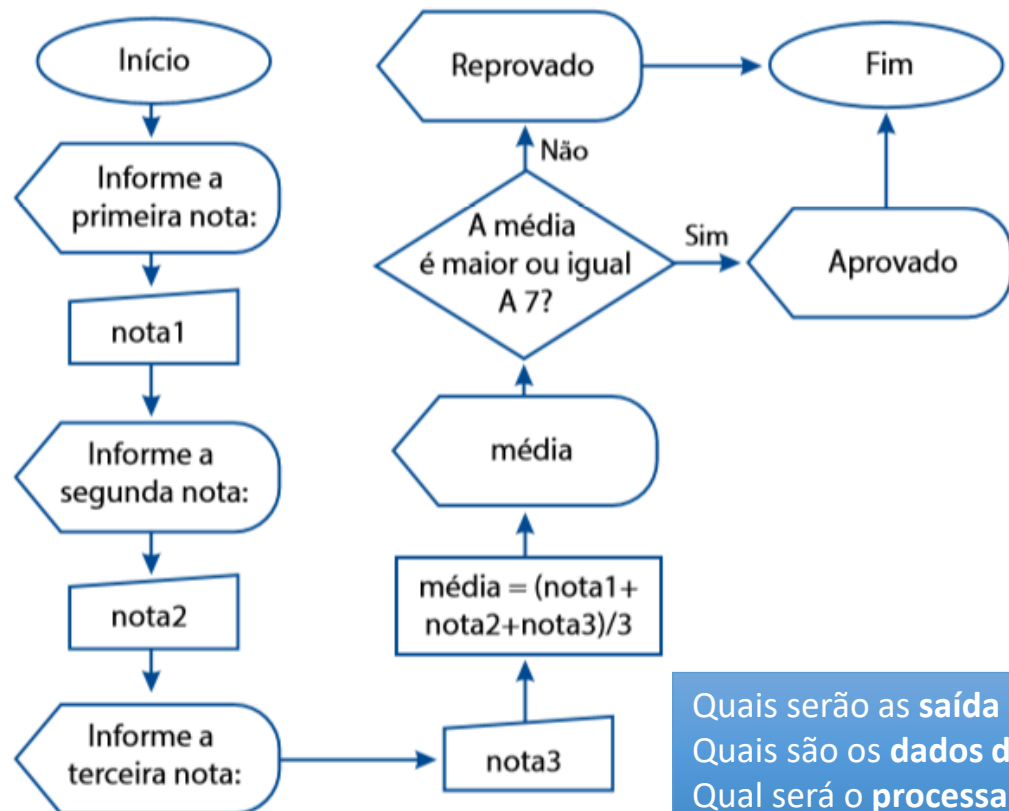
$$\text{Média} = \frac{\text{Nota1} + \text{Nota2} + \text{Nota3}}{3}$$



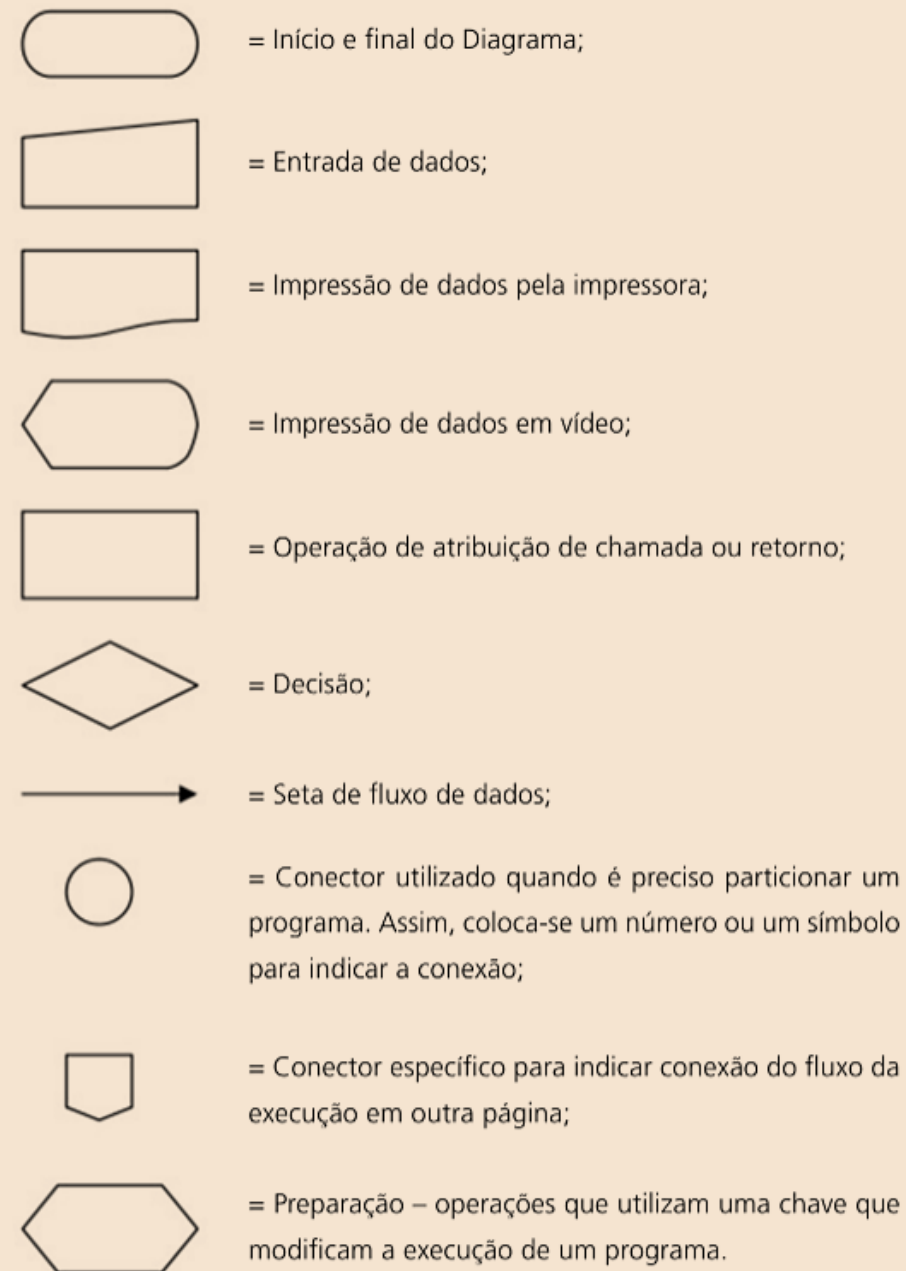
Representação de Algoritmos

ALGORITMO = Sequência lógica

FLUXOGRAMA OU DIAGRAMA DE BLOCOS



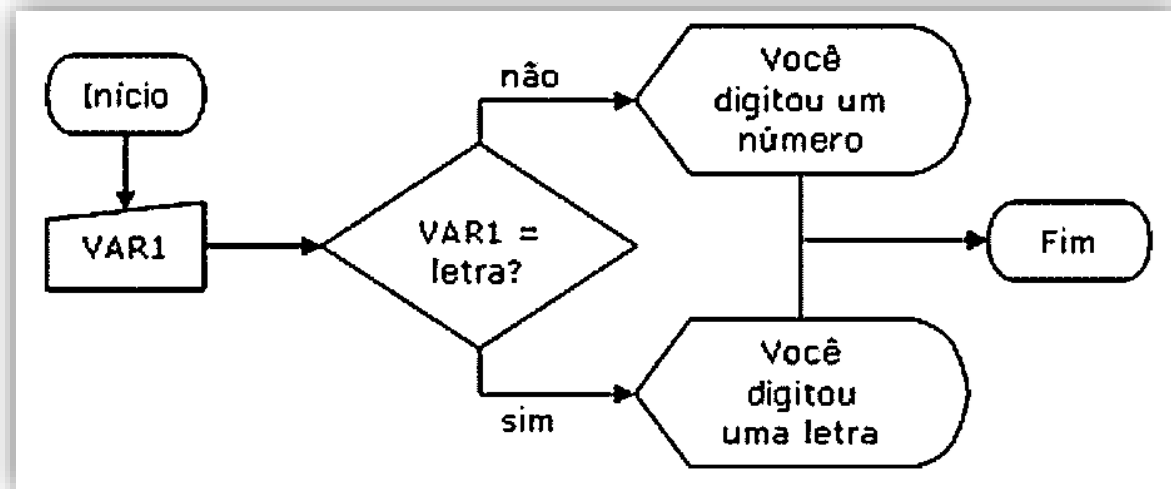
Quais serão as **saída de dados**?
Quais são os **dados de entrada**?
Qual será o **processamento** a ser utilizado?



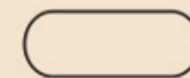
Representação de Algoritmos

ALGORITMO = Sequência lógica

FLUXOGRAMA OU DIAGRAMA DE BLOCOS



Quais serão as **saída de dados**?
Quais são os **dados de entrada**?
Qual será o **processamento** a ser utilizado?



= Início e final do Diagrama;



= Entrada de dados;



= Impressão de dados pela impressora;



= Impressão de dados em vídeo;



= Operação de atribuição de chamada ou retorno;



= Decisão;



= Seta de fluxo de dados;



= Conector utilizado quando é preciso particionar um programa. Assim, coloca-se um número ou um símbolo para indicar a conexão;



= Conector específico para indicar conexão do fluxo da execução em outra página;



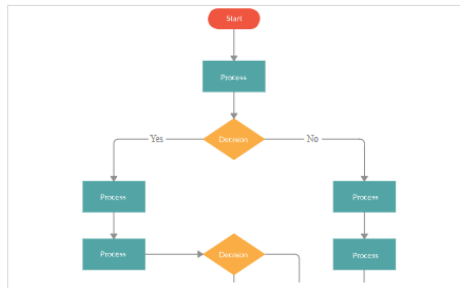
= Preparação – operações que utilizam uma chave que modificam a execução de um programa.

FLUXOGRAMA OU DIAGRAMA DE BLOCOS

Site para montar fluxograma online: <https://creately.com/>

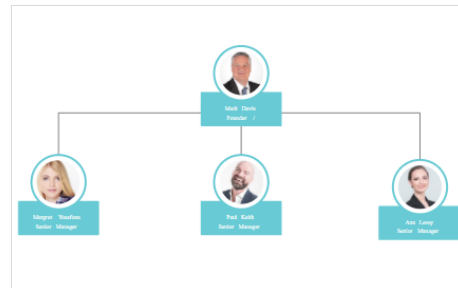
Mais de 50 bibliotecas especializadas de formas

Velocidade * é * um recurso. Qualquer esboço, conceito, análise, desenha no Creately várias vezes mais rápido



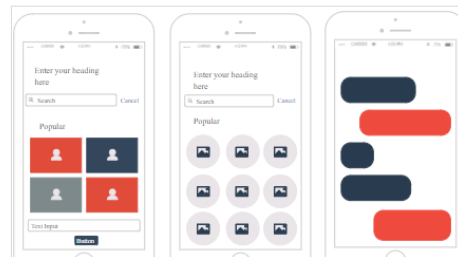
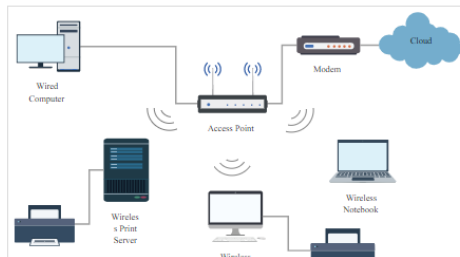
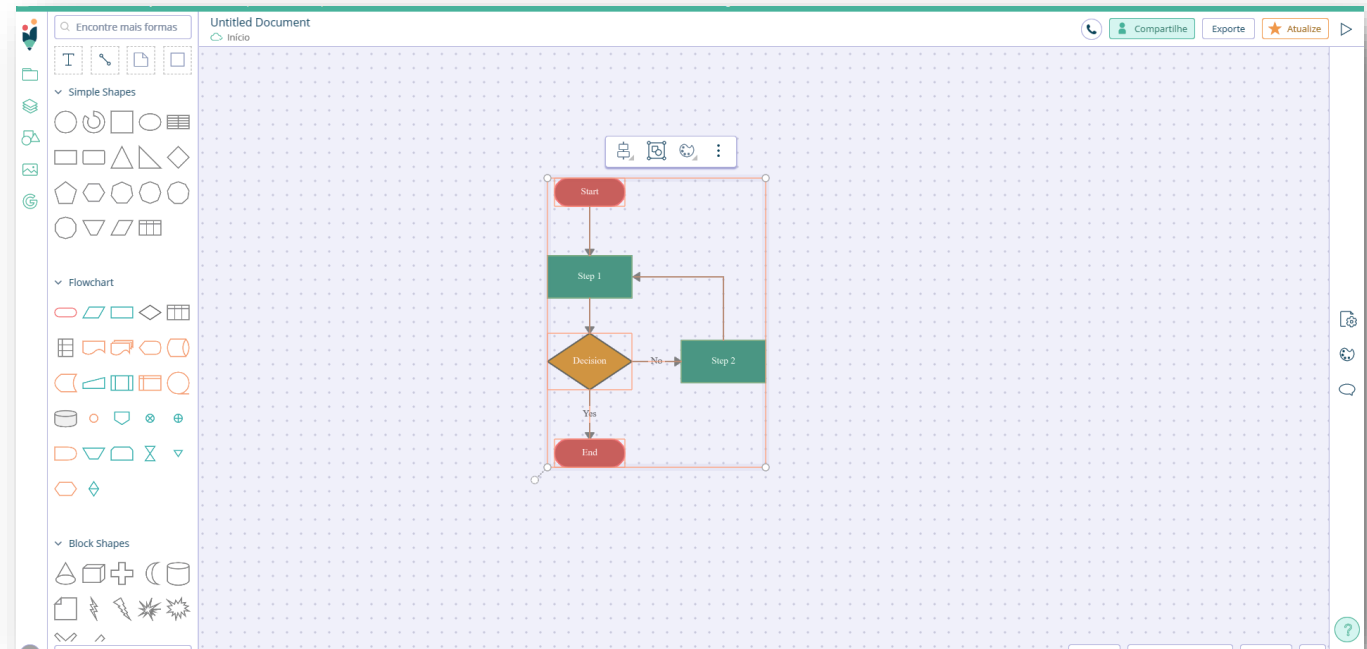
Fluxogramas

Maneira mais rápida de desenhar fluxogramas. Fluxos de conexão automática, raias e ícones adicionais.



Organogramas

Inclua imagens e adicione texto personalizado para projetar qualquer tipo de organograma.



Representação de Algoritmos

ALGORITMO = Sequência lógica

PSEUDOCÓDIGOS

Algoritmo <nome_do_algoritmo>

<declaração_de_variáveis>

<subalgoritmos>

Início

<corpo_do_algoritmo>

Fim.

Comandos básicos:

ESCREVA ou ESCREVER: Comando para exibir ao usuário um texto ou valor.

LEIA ou LER: Comando que receberá um valor digitado pelo usuário atribuindo a uma variável criada.

1- Faça um Programa que mostre a mensagem "Alo mundo" na tela.

```
ALGORITMO Programa_Alo
INICIO
    ESCREVER "Alo mundo";
FIM.
```

2- Faça um Programa que leia o nome do usuário mostre a mensagem "Olá <nome_do_usuario>"

```
ALGORITMO OlaUsuario
VAR nome : literal
INICIO
    ESCREVER "Qual é o seu nome? ",
    LER nome;
    ESCREVER "Olá ", nome;
FIM.
```

Representação de Algoritmos

ALGORITMO = Sequência lógica

PSEUDOCÓDIGOS

Algoritmo <nome_do_algoritmo>

<declaração_de_variáveis>

<subalgoritmos>

Início

<corpo_do_algoritmo>

Fim.

3- Faça um Programa que peça dois números e imprima a soma.

ALGORITMO Programa_Soma

VAR num1, num2, soma : INTEIRO;

INICIO

num1<- 0;

num2<- 0;

soma <- 0;

ESCREVER "Digite o primeiro número : ";

LER (num1);

ESCREVER "Digite o segundo número : ";

LER (num2);

soma <- num1 + num2;

ESCREVER "A soma é: ", soma;

FIM.

Representação de Algoritmos

ALGORITMO = Sequência lógica

PSEUDOCÓDIGOS

Algoritmo <nome_do_algoritmo>

<declaração_de_variáveis>

<subalgoritmos>

Início

<corpo_do_algoritmo>

Fim.

Algoritmo Média

Var N1, N2, Média

Início

Leia N1, N2

$Média := (N1 + N2) / 2$

Se $Média \geq 7$

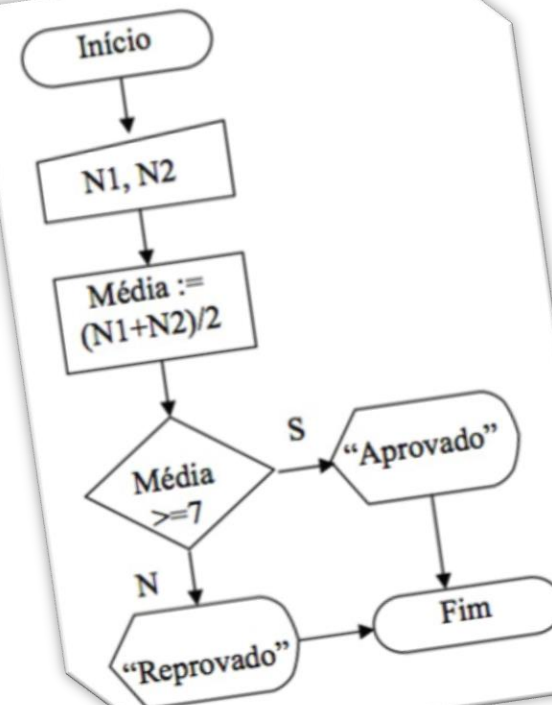
Então

Escreva "Aprovado"

Senão

Escreva "Reprovado"

Fim.



Representação de Algoritmos

ALGORITMO = Sequência lógica

PSEUDOCÓDIGOS

Algoritmo <nome_do_algoritmo>

<declaração_de_variáveis>

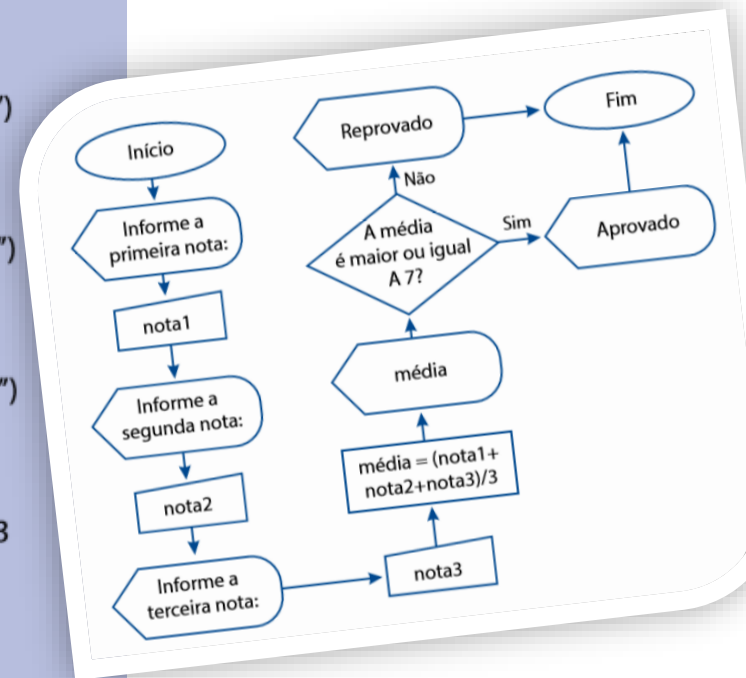
<subalgoritmos>

Início

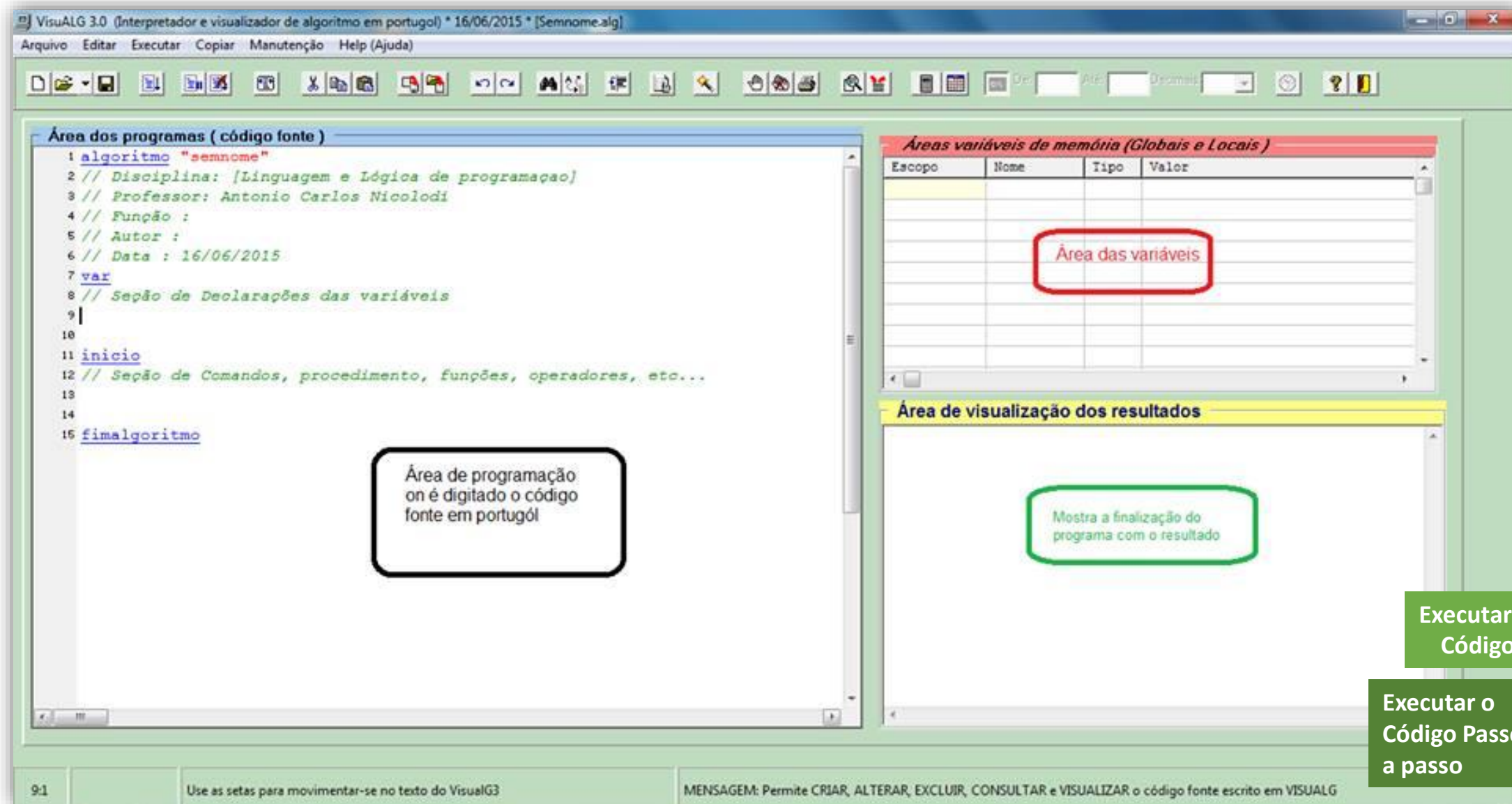
<corpo_do_algoritmo>

Fim.

1. algoritmo CalcularMedia
2. var
3. nota1, nota2, nota3, media : real
- 4.
5. início
6. escrever ("Digite a primeira nota:")
7. ler (nota1)
8. escrever ("Digite a segunda nota:")
9. ler (nota2)
10. escrever ("Digite a terceira nota:")
11. ler (nota3)
12. $media \leftarrow (nota1 + nota2 + nota3) / 3$
13. escrever (media)
14. se $media \geq 7$ então
15. escrever ("Aprovado")
16. senão
17. escrever ("Reprovado")
18. fimse
19. fim.



O VisualG <http://visualg3.com.br>



Executar o
Código

F9

Executar o
Código Passo
a passo

F8

O VisualG

```
Área dos algoritmos ( Edição do código fonte ) -> Nome do arquivo: [semnome]

1 Algoritmo "semnome"
2 // Disciplina   : [Linguagem e Lógica de Programação]
3 // Professor    : Antonio Carlos Nicolodi
4 // Descrição    : Aqui você descreve o que o programa faz! (função)
5 // Autor(s)     : Nome do(a) aluno(a)
6 // Data atual   : 10/10/2017
7 Var
8 // Seção de Declarações das variáveis
9
10
11 Inicio
12 // Seção de Comandos, procedimento, funções, operadores, etc...
13
14
15 Fimalgoritmo
```

Comentários:

// Usando duas barras posso comentar na maioria das linguagens de programação
//Comentários são ignorados pelo interpretador

```
/* Comentário
Com
Mais de
Uma linha
*/
```

←Outras linguagens

BOAS PRÁTICAS: Comentários

- Início do arquivo/método indicando o objetivo do mesmo através de um pequeno resumo
- Descrevendo a especificação do código
- Não poluir o código, usar textos curtos e apenas os importantes/relevantes
- Pensar na leitura de outra pessoa

Algoritmo <nome_do_algoritmo>

<declaração_de_variáveis>

<subalgoritmos>

Início

<corpo_do_algoritmo>

Fim.

O VisualG *Leitura e Escrita de Dados*

Pseudocódigo

```
ALGORITMO Programa_Alo
INICIO
    ESCREVER "Alo mundo"
FIM.
```

```
ALGORITMO OlaUsuario
VAR nome : literal
INICIO
    ESCREVER "Qual é o seu nome?"
    LER nome
    ESCREVER "Olá ", nome
FIM.
```

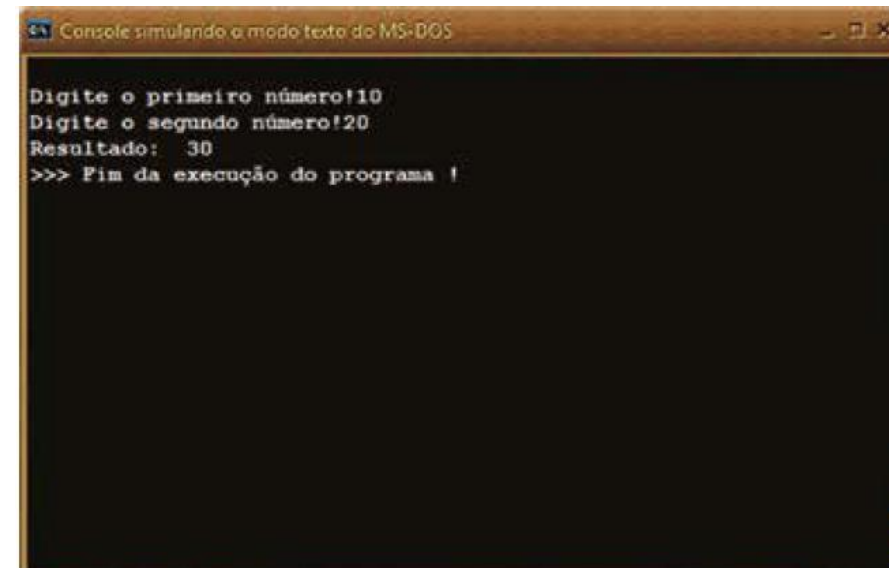
Sintaxe no VisualG

leia (variável) **escreva** (" ")
 escreva (" ", variável)
 escreval (" ", variável)

VisualG

```
7 Var
8 num1, num2, soma : inteiro
9
10 Inicio
11 escreva ("Digite o primeiro número!")
12 leia (num1)
13 escreva ("Digite o segundo número!")
14 leia (num2)
15 soma <- num1 + num2
16 escreva ("Resultado: ", soma)
17
18 Fimalgoritmo
```

Entradas do usuário: num1 = 10, num2 = 20



```
Console simulando o modo texto do MS-DOS
Digite o primeiro número!10
Digite o segundo número!20
Resultado: 30
>>> Fim da execução do programa !
```

O VisualG

Vamos praticar?
Resolução

1- Faça um Programa que mostre a mensagem "Alo mundo" na tela no VisualG.

Pseudocódigo

```
ALGORITMO Programa_Alo  
INICIO  
    ESCREVER "Alo mundo"  
FIM.
```

```
1 Algoritmo "Programa_Alo"  
2 Var  
3  
4 Inicio  
5     escreva ("Alo mundo")  
6 Fimalgoritmo
```



Console simulando o modo texto do MS-DOS

```
Alo mundo  
>>> Fim da execução do programa !
```


O VisualG

Vamos praticar?

Resolução

Pseudocódigo

```
ALGORITMO OlaUsuario
VAR nome : literal
INICIO
    ESCREVER "Qual é o seu nome? "
    LER nome
    ESCREVER "Olá ", nome
FIM.
```

2- Faça um Programa que leia o nome do usuário mostre a mensagem "Olá <nome_do_usuario>"

```
1 Algoritmo "OlaUsuario"
2 Var
3 nome: caracter
4
5 Inicio
6     escreval("Qual é o seu nome?")
7     leia(nome)
8
9     escreval("")
10    escreva ("Olá ", nome)
11 Fimalgoritmo
```

C:\ Console simulando o modo texto do MS-DOS

```
Qual é o seu nome?
Joãozinho Cabral de Vera Cruz

Olá Joãozinho Cabral de Vera Cruz
>>> Fim da execução do programa !
```

CONSTRUÇÃO DE UM BOM PSEUDOCÓDIGO OU ALGORITMO

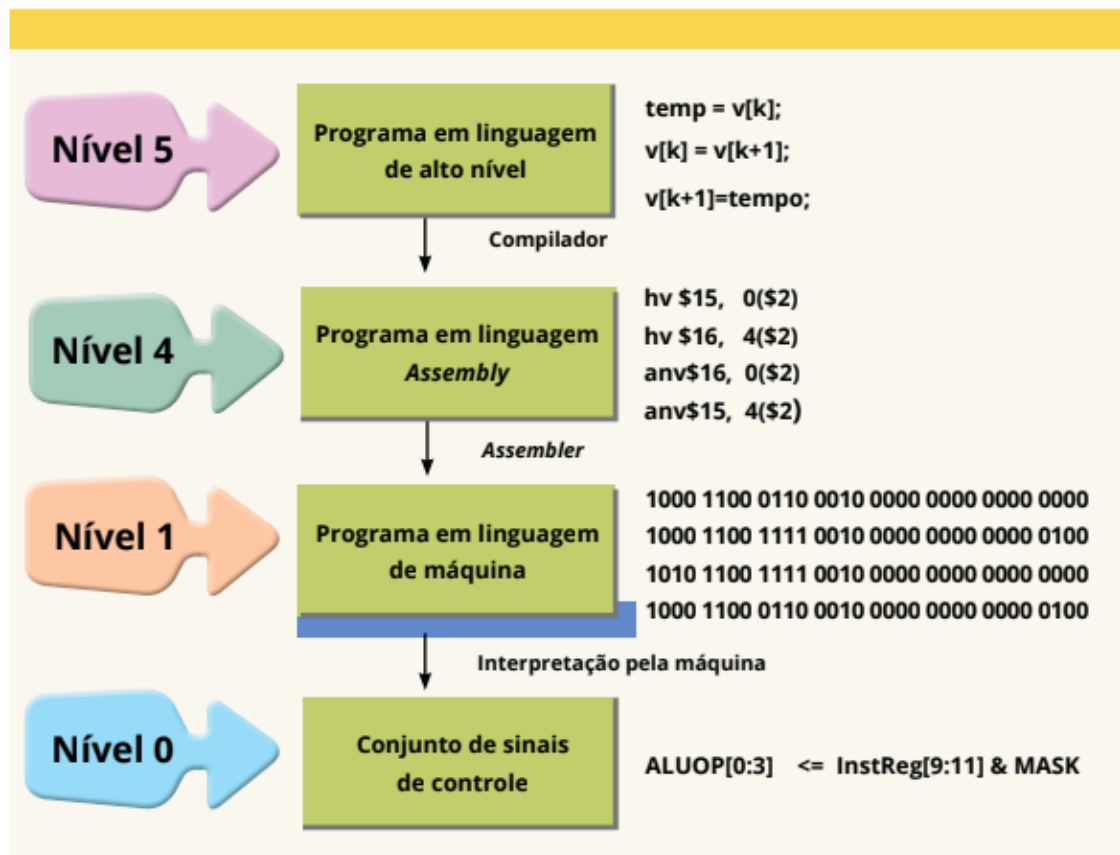
Para se escrever um bom pseudocódigo e tentar descrever uma sequência lógica de um programa, de forma que seja objetiva na finalidade de interpretar uma ação computacional por meio de uma linguagem, (no caso o português ou português estruturado) levaríamos em conta os seguintes detalhes abaixo:

1. Usar apenas verbos na fase de escrita;
2. Imaginar que a escrita do pseudocódigo está direcionada a pessoas que não conhecem de informática.
3. Usar frases curtas e diretas;
4. Ser claro e objetivo;
5. Tomar cuidado com palavras que possam ser encaradas com duplo sentido.



Linguagem de Programação

Figura - Codificação de programas e "níveis" na arquitetura de computadores



Nível 5 - Linguagem de Alto Nível: A = 5;

Nível 4 - Linguagem *Assembly*: MOVE A, #5;

Nível 0 - Linguagem de Máquina: 0011001100000101



LINGUAGEM INTERPRETADA

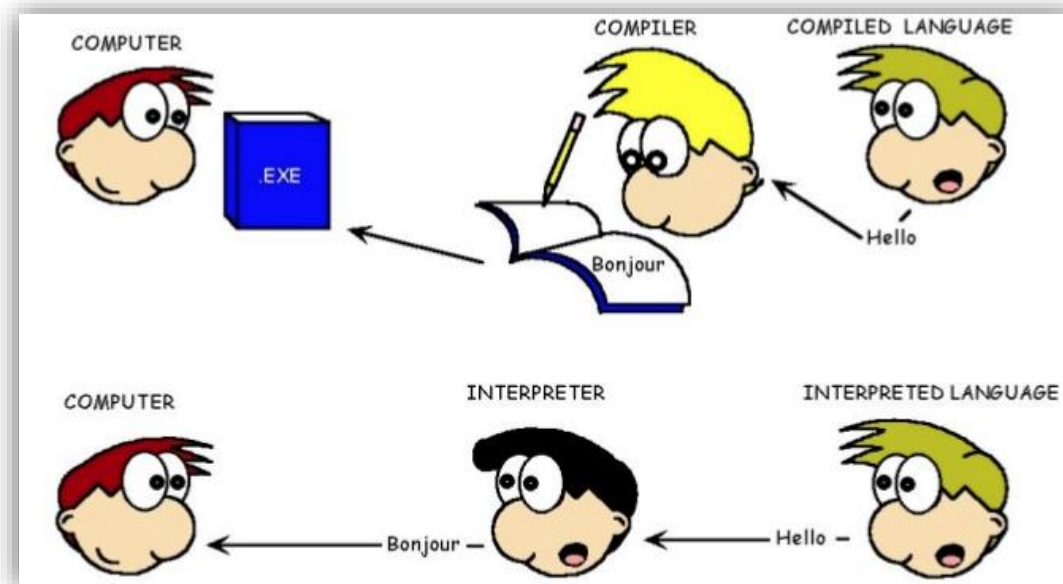
- Executam o código do programa à medida que este vai sendo traduzido → mais flexível
- Ex: JavaScript, e Python.

Java e C#



LINGUAGEM COMPILADA

- Traduzem todo o código do programa para apenas depois executar → **compilador** → tradução para a linguagem de máquina.
- Ex: C e C++



Lógica de Programação

Linguagem de Programação



Código fonte

```
#include <iostream.h>
#include <conio.h>

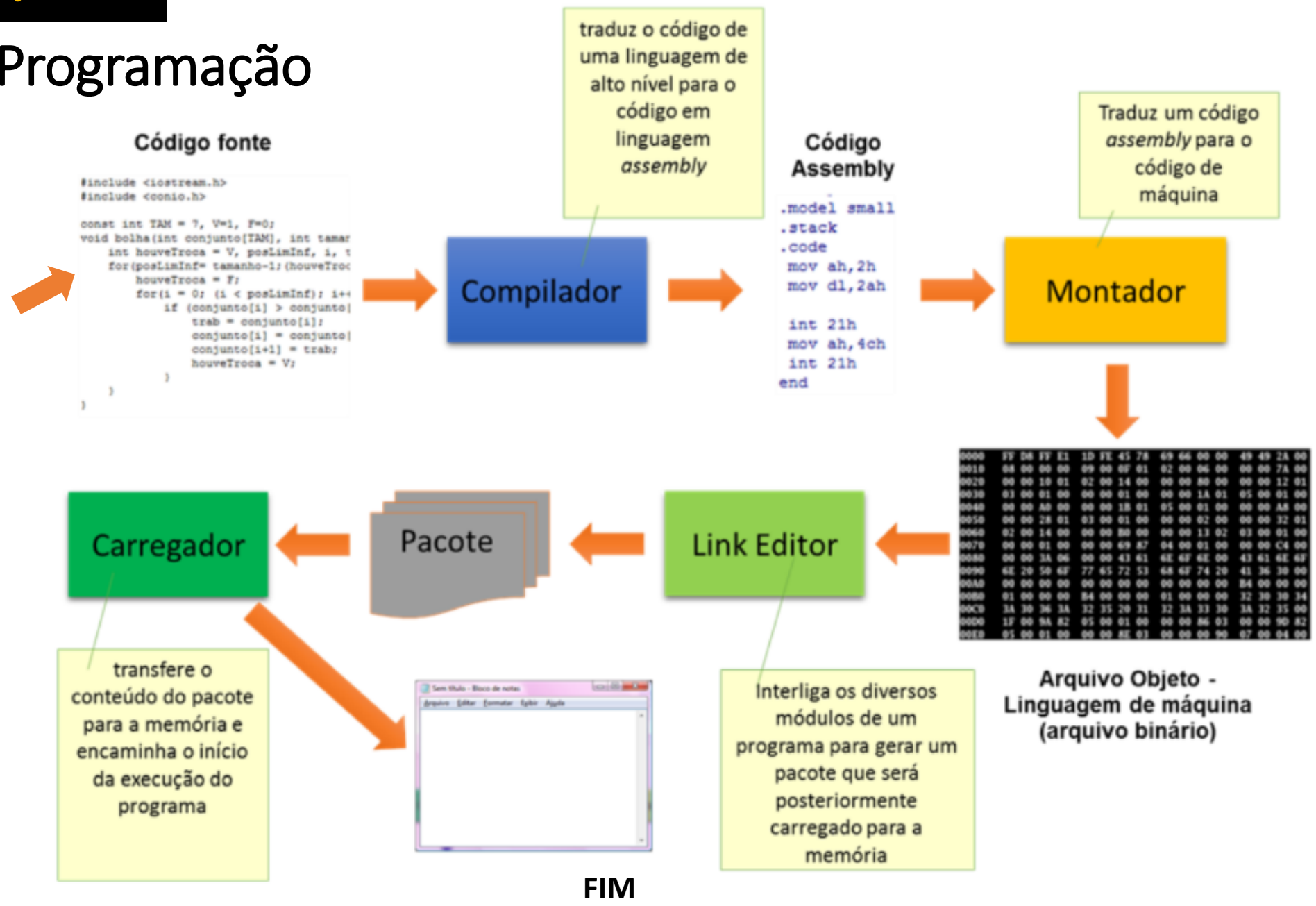
const int TAM = 7, V=1, F=0;
void bolha(int conjunto[TAM], int tamar
int houveTroca = V, posLimInf, i, t
for(posLimInf= tamanho-1; (houveTroc
houveTroca = F;
for(i = 0; (i < posLimInf); i++
if (conjunto[i] > conjunto[i+1])
trab = conjunto[i];
conjunto[i] = conjunto[i+1];
conjunto[i+1] = trab;
houveTroca = V;
}
```

- ☒ Entender o Problema
- ☐ Requisitos, entradas e saídas
- ☐ Fluxograma e Pseudocódigo
- ☐ Algoritmos

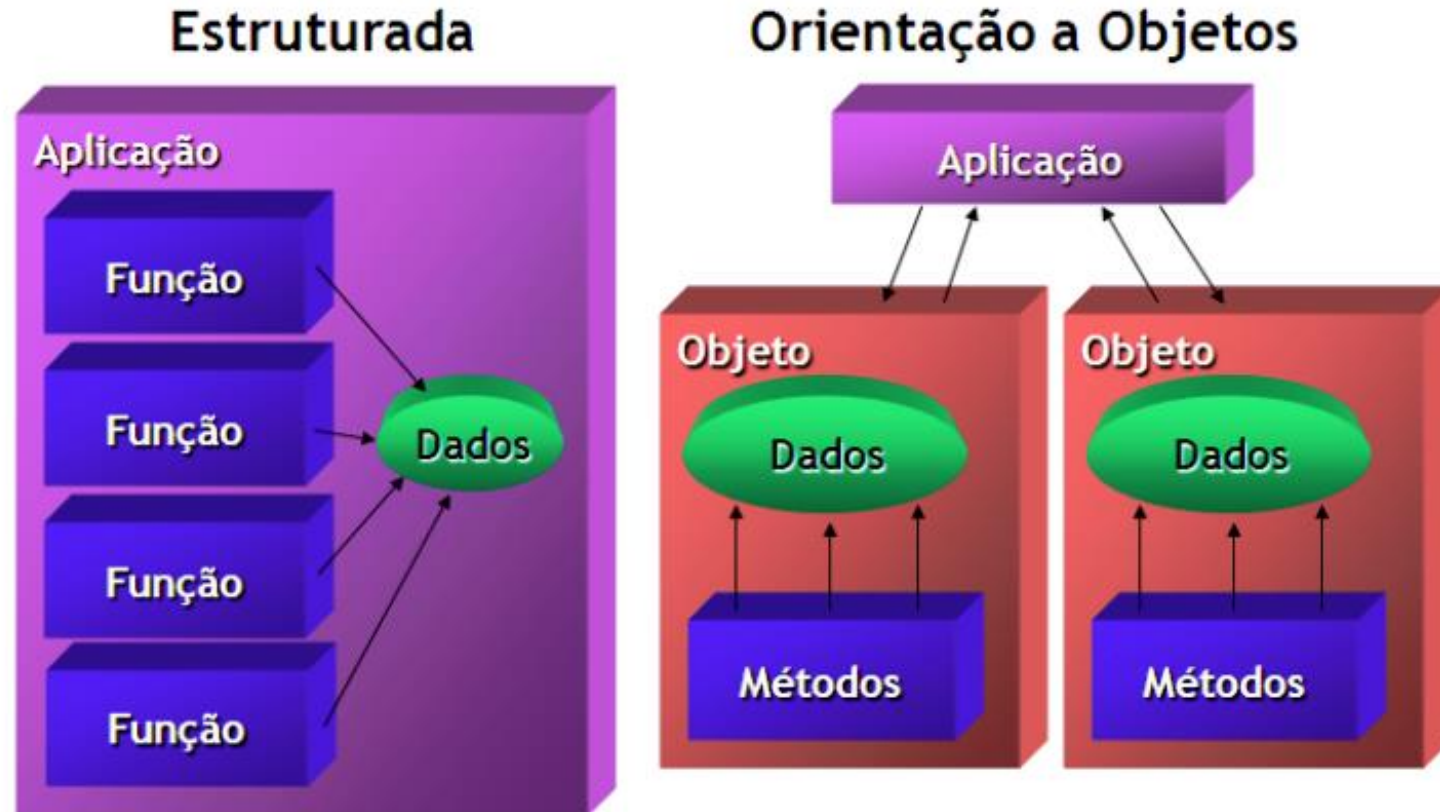


INICIO

Passos para a criação de um programa



Linguagem de Programação



Qual linguagem de programação escolher?

<https://www.youtube.com/watch?v=DKBC5pEpCLY&list=PL99AE7FAAE4560E2A&index=4>

Estudando Lógica de Programação

Algoritmos **não** se aprendem
Copiando algoritmos / Estudando algoritmos

Algoritmos **só se aprendem**
Construindo algoritmos / Testando algoritmos



Lista de exercícios – Estrutura Sequencial
<https://wiki.python.org.br/EstruturaSequencial>



Entender o Problema



Requisitos, entradas e saídas



Fluxograma e Pseudocódigo



Algoritmos



Pseudocódigo sem refinamento? Exemplos:

ALGORITMO INICIAL (sem refinamento)

Início

Receber a NOTA 1;
Receber a NOTA 2;
Receber a NOTA 3;
Receber a NOTA 4;
Somar todas as notas de entrada e dividir o resultado por 4 (quatro);
Mostrar o resultado final, o qual foi armazenado em Média;

fim.

Algoritmo TELA INICIAL (Refinar)

INICIO

Mostrar opções : Novo Jogo ou Sair

Se escolher SAIR,

Escrever: "Obrigado por jogar"

Fechar jogo ()

SE escolher NOVO JOGO,

Iniciar Partida.

FIM

ALGORITMO NOVO JOGO (REFINAR)

INICIO

Enquanto a tecla for diferente de PAUSE FAÇA

Repita

Mostrar uma pergunta

SE usuário acertar: Adicionar 10 pontos a sua pontuação

SE usuário errar, mostrar na tela "Infelizmente você perdeu",
mostrar a pontuação e VOLTAR para a TELA INICIAL

SE usuário acertar todas as questões: mostrar na tela:
"Parabéns você é um campeão!!!" e VOLTAR para a TELA
INICIAL

até que usuário ERRE UMA QUESTÃO

FIM ENQUANTO

FIM

Todo mundo deveria aprender a programar (What most schools don't teach - dublado):

<https://www.youtube.com/watch?v=4iKu9qtCSXg>

O QUE POSSO FAZER COM PROGRAMAÇÃO?

https://www.youtube.com/watch?v=EvHeuJs_hcQ

Afinal, o que é uma linguagem de programação?

<https://www.youtube.com/watch?v=PmefpISZ7Ew&list=PL99AE7FAAE4560E2A&index=1>

Afinal, o que é um algoritmo?

<https://www.youtube.com/watch?v=yTKiRkCi0Bg&index=3&list=PL99AE7FAAE4560E2A>

Lógica de Programação com VisualG- **Prof. Thiago de Oliveira** (Vários conteúdos para estudo e aprofundamento)

<https://sites.google.com/a/ifmg.edu.br/troliveira/disciplinas/logica-de-programacao>

Nesse vídeo temos dois exercícios básicos feitos e simulados no VisualG:

<https://www.youtube.com/watch?v=mz7xFwwSZts>



Aula 2 - Introdução à programação de computadores - parte1

<https://pt.slideshare.net/henriquecarmona/introduo-programao-de-computadores-parte1>

Aula 2 - Introdução à programação de computadores - parte 2

<https://pt.slideshare.net/henriquecarmona/aula-2-introduo-programao-de-computadores-parte2>