



LÓGICA DE PROGRAMAÇÃO

+55 (21) 99461-8818

@explicadoresnet

www.explicadores.net.br



Abordagem Contextual

RACIOCÍNIO
+
LÓGICA

Raciocínio lógico

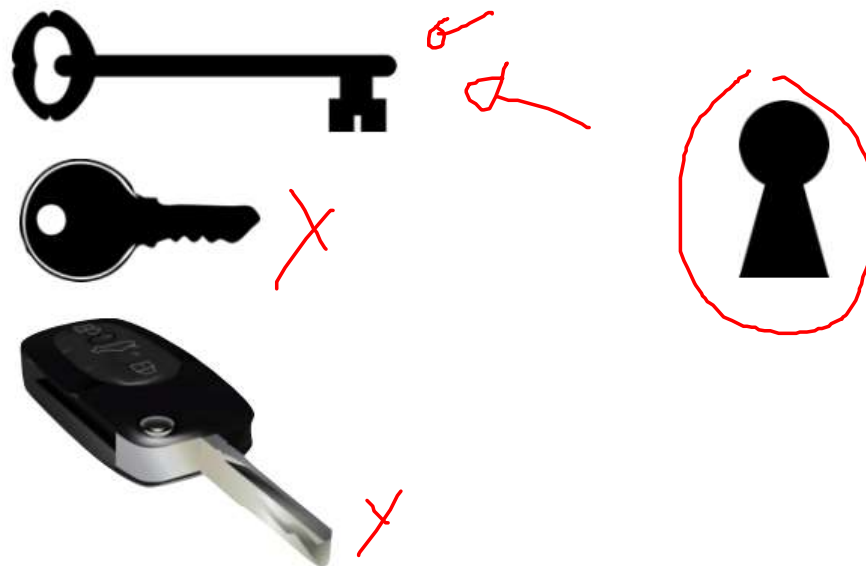
Para usar a **lógica**, é necessário ter domínio sobre o **pensamento**, bem como saber pensar, ou seja, possuir a “Arte de Pensar”.

É um **esquema sistemático** que define as interações de sinais no equipamento automático do processamento de dados, ou o computador científico com o critério e princípios formais de **raciocínio** e **pensamento**.

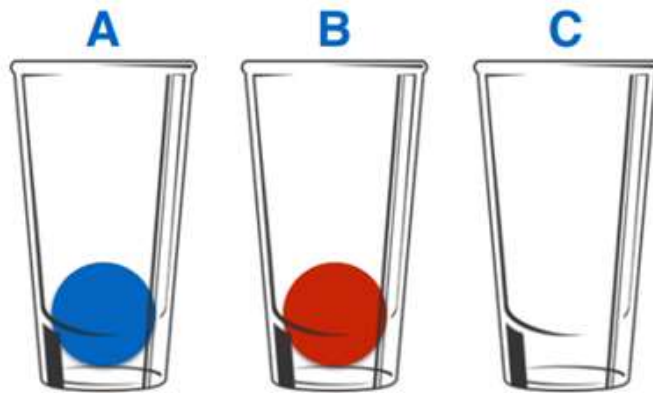
Usar a **lógica** é um fator a ser considerado por todos, principalmente pelos profissionais da área da Tecnologia de Informação.

Conceitos Fundamentais

“**Lógica** é um conjunto de princípios e de métodos que embasam raciocínios coerentes, induzindo a soluções eficazes para problemas.”



Raciocínio Lógico



Coloque ● em C
Coloque ● em A
Coloque ● em B

Conteúdo de A vai para C
Conteúdo de B vai para A
Conteúdo de C vai para B

$C \leftarrow A$
 $A \leftarrow B$
 $B \leftarrow C$

Isso é **ALGORITMO!**

Aplicabilidade da Lógica na Programação

A técnica mais importante no **projeto da lógica de programas** é chamada **programação estruturada**, a qual consiste em uma metodologia de projeto, objetivando:

- **agilizar** a codificação da escrita de programas;
- **facilitar** a **depuração** da sua leitura;
- permitir a **verificação** de possíveis **falhas** apresentadas pelos programas;
- **facilitar** as **alterações** e atualizações dos programas.

ESTRUTURAS LÓGICA
SEQUÊNCIA
SELEÇÃO / DESVIO
ITERAÇÃO → REPETIÇÃO
LOOP

→ BUSCAR ERROS!



Aplicabilidade da Lógica na Programação

E deve ser composta por **quatro passos fundamentais**:

- Escrever as instruções em sequências ligadas entre si apenas por **estruturas sequenciais**, **repetitivas** ou de **seleção**.
- Escrever instruções em **grupos pequenos** e combiná-las.
- Distribuir **módulos** do programa entre os diferentes programadores que trabalharão sob a supervisão de um **programador sênior**, ou chefe de programação.
- Revisar o trabalho executado em **reuniões regulares** e previamente programadas, em que compareçam programadores de um mesmo nível.

INSTRUÇÃO
- LEIA A, B ✓
- C ← A + B ✓
- ESCREVA C ✓

PROGRAMA -
- CONJUNTO DE INSTRUÇÕES
FUNÇÃO/MÓDULO / SUB-ROTIMA
- GRUPO DE INSTRUÇÕES
MÓDULOS



Conceito

“**Algoritmo** é uma descrição das etapas de resolução de um problema ou a indicação ordenada de uma sequência de ações bem definidas.”

“**Algoritmo** é a maneira mais elementar de descrever uma **lógica**.”

“**Algoritmo** é um processo de **cálculo matemático** ou de resolução de um grupo de problemas semelhantes. Pode-se dizer também que são regras formais para obtenção de um resultado ou da solução de um problema, englobando **fórmulas** de **expressões aritméticas**.”

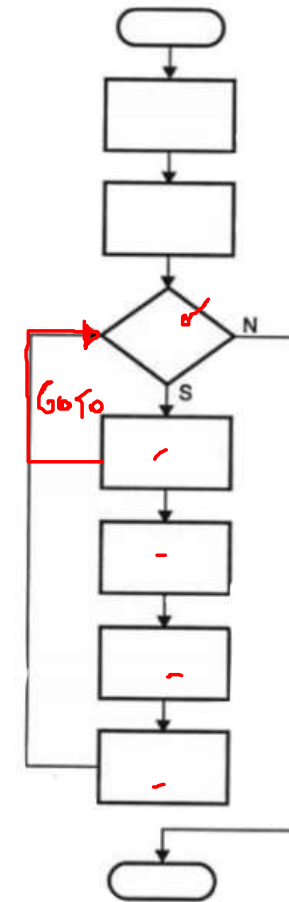


Tipos de Lógica

Lógica Linear

Coleção de elementos organizados ou ordenados por uma só propriedade, de tal forma que cada um deles seja executado passo a passo de cima para baixo, em que tenha um só “predecessor” e um só “sucessor”.

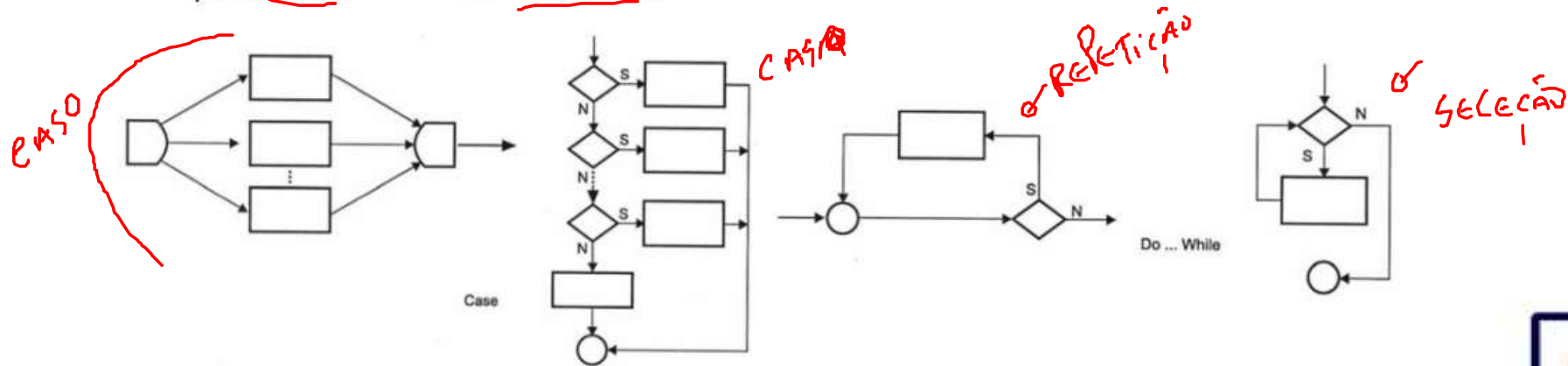
Não está ligada a regras de hierarquia ou de estruturas de linguagens específicas de programação de computadores.



Tipos de Lógica

Lógica Estruturada

É usada pelos profissionais de processamento eletrônico de dados. Possui características e padrões particulares, os quais diferem dos modelos das linguagens elaboradas por seus fabricantes. Tem como pontos fortes para elaboração futura de um programa, produzi-lo com alta qualidade e baixo custo.

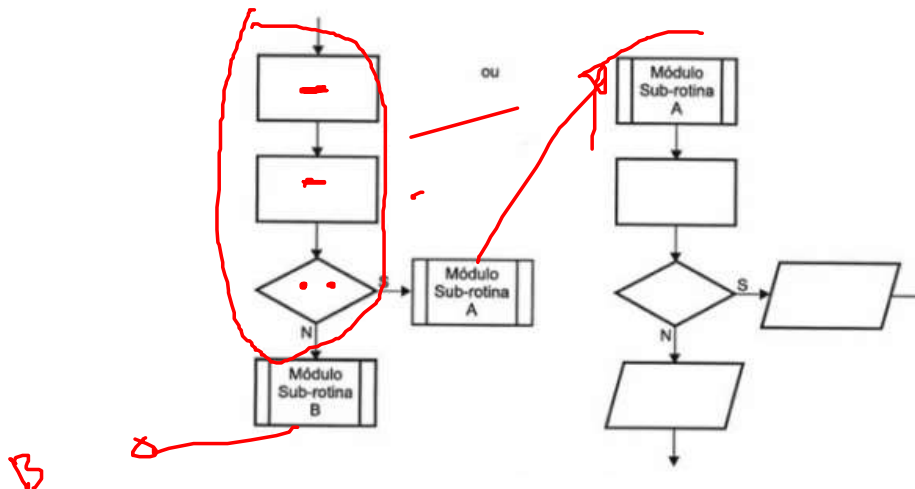
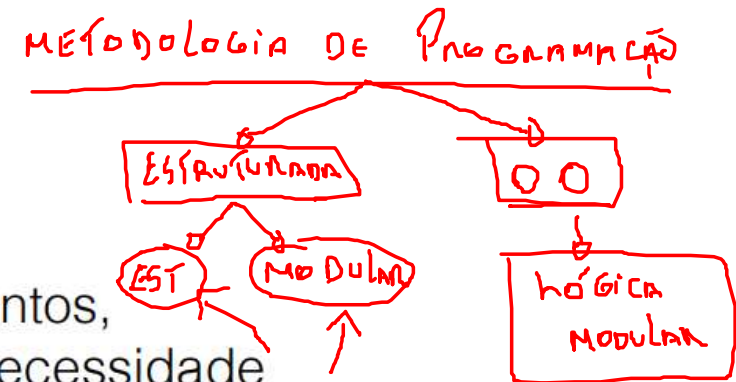


Tipos de Lógica

Lógica Modular

Soluciona problemas dividindo-os em segmentos, tratando de cada ação na ordem em que a necessidade for aparecendo.

Utilizada para separar um problema em sistemas, um sistema em programas e um programa em módulos.



Instrumentos da Lógica de Programação

Formato: Narrativo

Português Estruturado

Pseudocódigo

```
Fazer Enquanto tiver aluno
    Obter a matrícula
    Obter as notas
    Calcular a média
    Se média suficiente
        Aluno Aprovado
    senão
        Aluno Reprovado
Fim Fazer Enquanto
```

Esta técnica de algoritmização é baseada em uma PDL – Program Design Language (Linguagem de Projeto de programação).

A PDL (neste caso, o português estruturado) é usado como referência genérica para uma linguagem de projeto de programação, tendo como finalidade mostrar uma notação para elaboração de algoritmos, os quais serão utilizados na definição, criação e desenvolvimento em uma linguagem computacional (Cobol, Fortran, C, Pascal, Delphi, Visual-Basic, etc.).

Instrumentos da Lógica de Programação

Formato: Narrativo

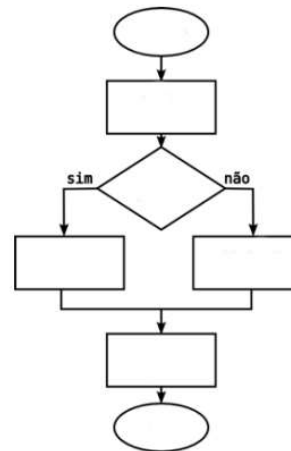
Portugol

```
variáveis
    Mat, N1, N2, M, Total, Cont
Leia(Total)
Cont <- 1
Enquanto (Cont <= Total)
    Leia (Mat, N1, N2)
    M <- (N1 + N2) / 2
    Se M >= 7 então
        Escreva("Aluno Aprovado")
    senão
        Escreva("Aluno Reprovado")
    FimSe
FimEnquanto
```

Instrumentos da Lógica de Programação

Formato: Gráfico

Diagrama de Blocos



Ou **diagrama de fluxo** é uma ferramenta usada e desenvolvida pelo profissional que está envolvido **diretamente com a programação**, tendo como objetivo descrever o método e a sequência do processo dos planos num computador.

Instrumentos da Lógica de Programação

Formato: Gráfico

Fluxograma

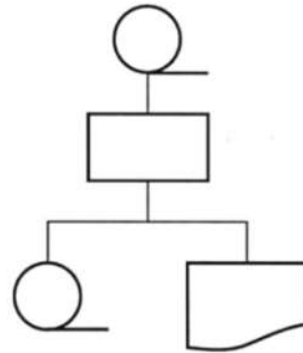


Figura 1.1 - Exemplo de um fluxograma.

Ferramenta usada e desenvolvida pelos profissionais de análise de sistemas, bem como, por alguns profissionais de **Organização, Sistemas e Métodos**. Tem como finalidade descrever o fluxo, seja manual ou mecânico.

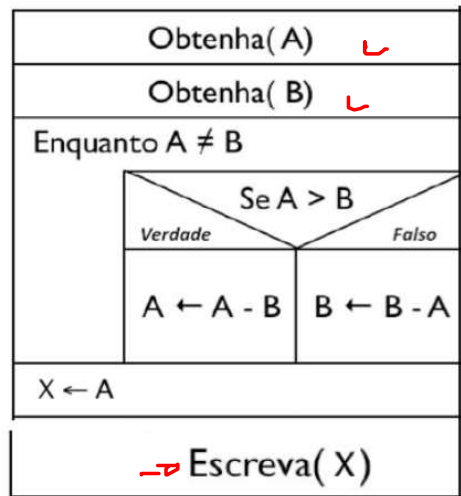
Instrumentos da Lógica de Programação

Formato: Gráfico

Diagrama de Chapin

Nassi-Shneiderman

Sequência
Repetição



Seleção

O diagrama foi desenvolvido por Nassi e Shneiderman e ampliado por Ned Chapin, os quais resolveram substituir o diagrama de blocos tradicional por um diagrama de quadros que permite apresentar uma visão hierárquica e estruturada da lógica do problema. A grande vantagem de usar este tipo de diagrama é a representação das estruturas que tem um ponto de entrada e um ponto de saída e são compostos pelas estruturas básicas de controle de sequência, seleção e repetição. Este tipo de diagrama também é denominado Diagrama de Shneiderman ou diagrama N-S.

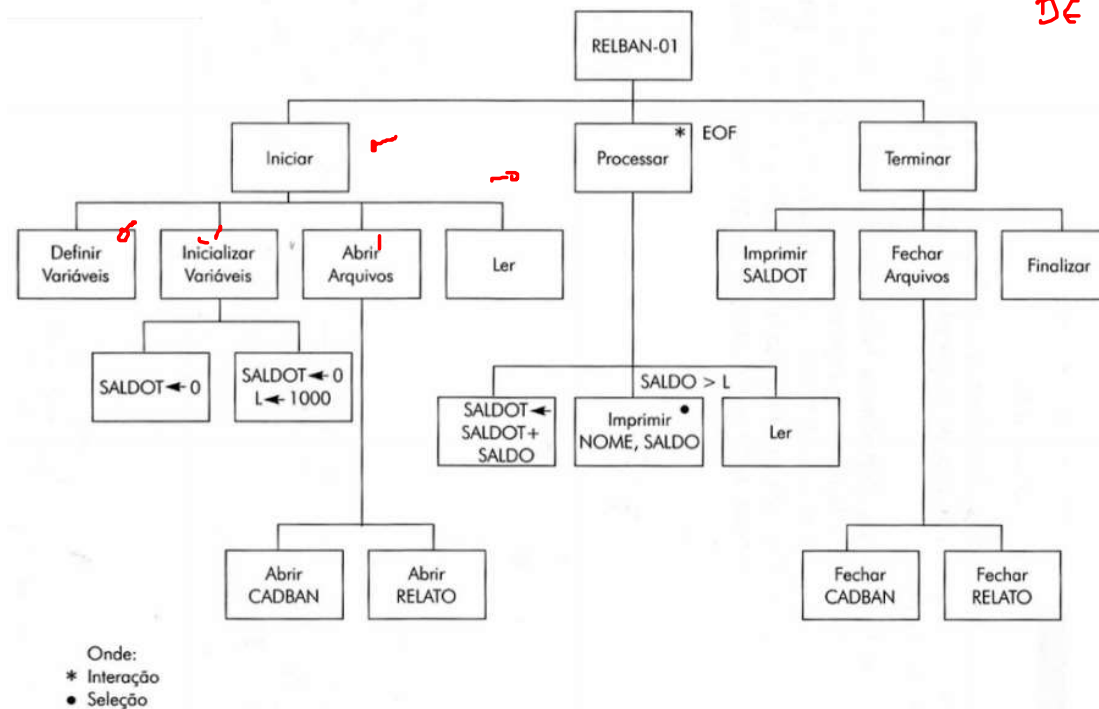


Instrumentos da Lógica de Programação

Formato: Em Detalhes

Diagrama Hierárquico de Fluxo

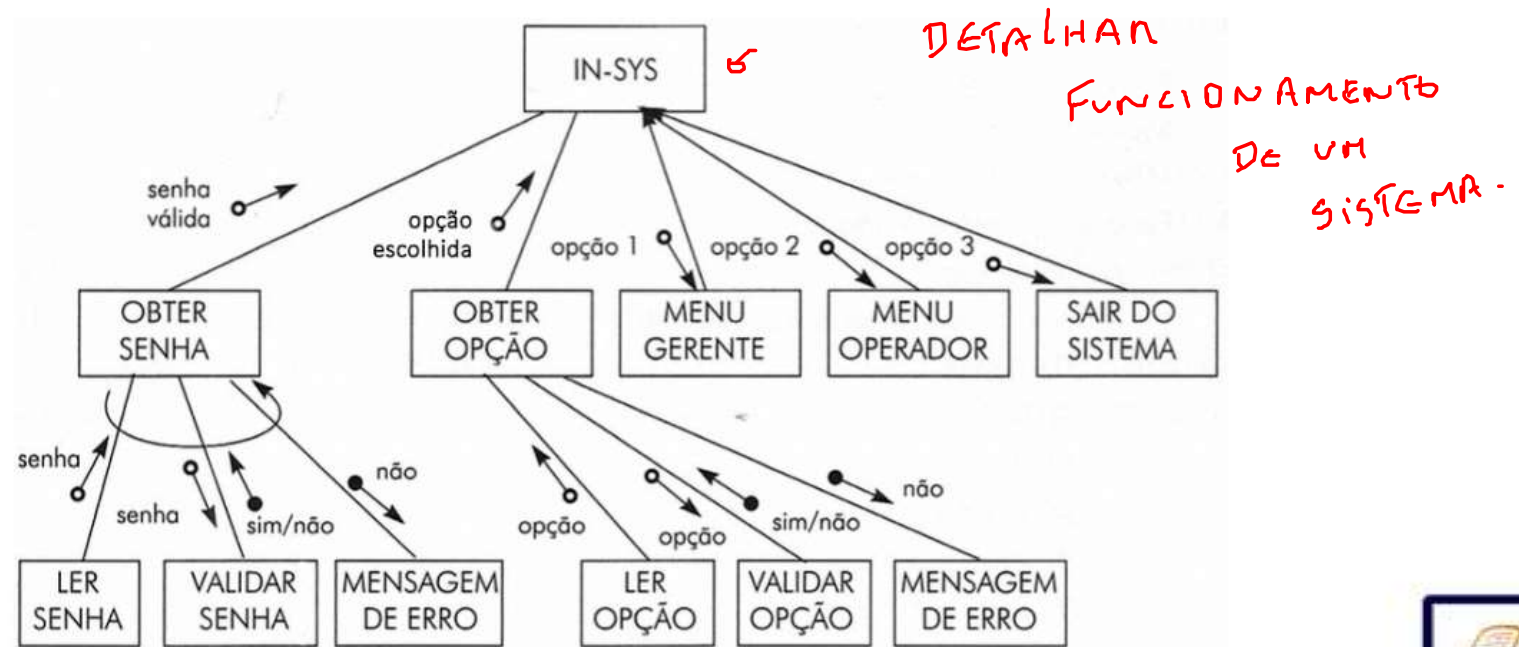
DETALHAM AS FUNCIONALIDADES
DE UM SISTEMA!



Instrumentos da Lógica de Programação

Formato: Em Detalhes

Gráfico de Estruturas (GRAFO)



Instrumentos da Lógica de Programação

Formato: Em Detalhes

Árvore de Decisões

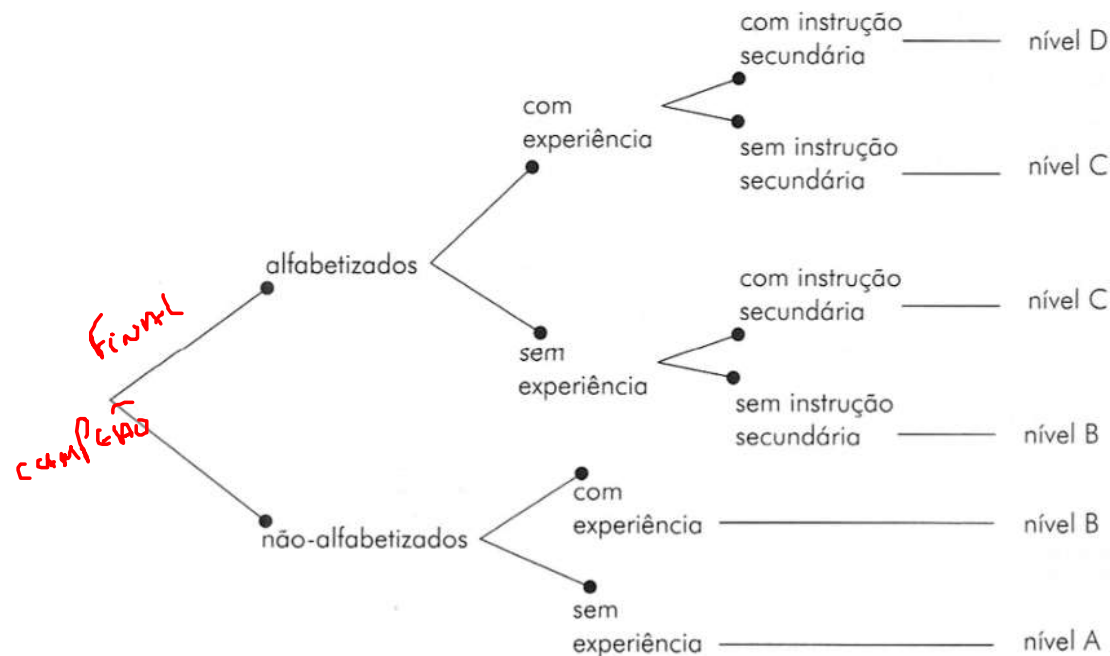


Diagrama de Blocos: Simbologia

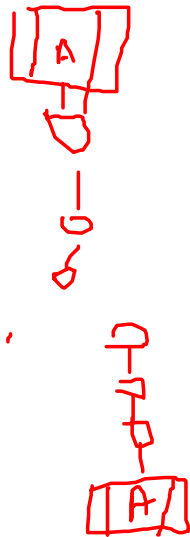
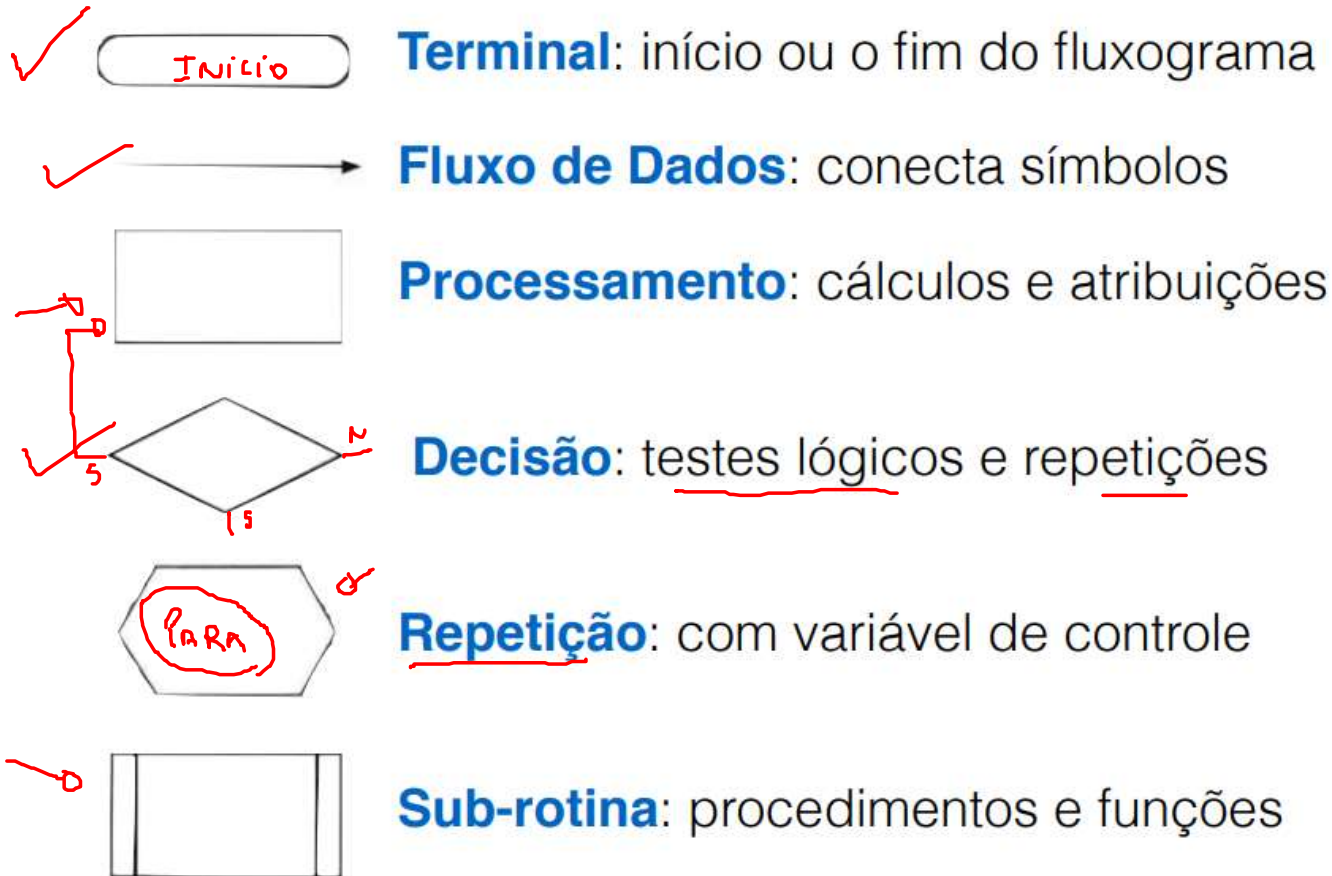


Diagrama de Blocos: Simbologia

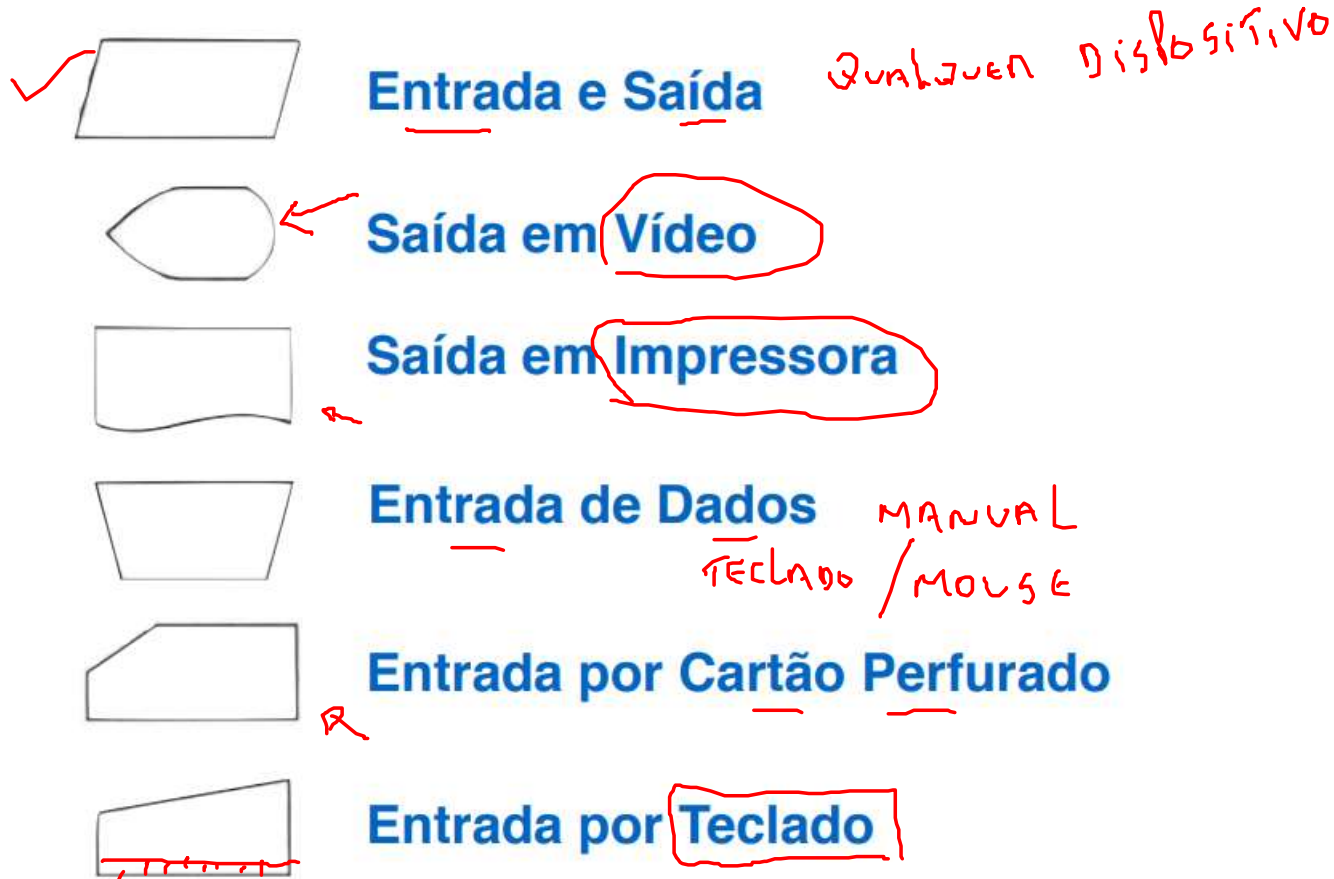
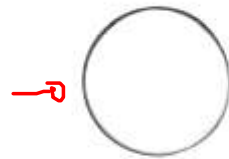
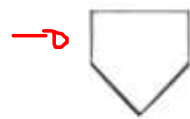


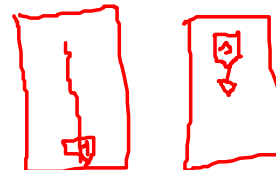
Diagrama de Blocos: Simbologia



Conector de Fluxo



Conector de Página



Abaixo seguem outros símbolos normalmente usados em outras áreas, bem como também pela área de desenvolvimento de software.



	Operação manual - fora de linha sem intervenção de dispositivos eletromecânicos.
	Modificação de programas - indica a existência de uma instrução ou de um grupo de instruções que irão modificar o programa, o qual poderá estar descrito ou em análise.
	Cartão perfurado - todas as variedades apresentadas. Essa massa de cartões poderá ser usada como documentos escritos anteriormente.
	Preparação - refere-se a um determinado grupo de operações não incluídas na diagramação, bem como, na elaboração de uma chave que modificará a execução de um determinado programa.

Preparação
Preparação

R

	Teclado - serão as informações recebidas ou fornecidas de ou por um computador.
	Display - para informações exibidas por dispositivos visuais, vídeo ou monitor.
	Checagem de operação auxiliar - é uma operação de máquina suplementar à função principal de processamento.
	Disco magnético - memória de massa para armazenamento de dados (winchester).
	Tambor magnético - memória de massa para armazenamento de dados.
	Fita magnética - memória de massa para armazenamento de dados (streamer).
	Disquete - memória de armazenamento de dados.
	Linha de comunicação - permite a transmissão automática de informação em locais diferentes, por meio de linhas de comunicação.

Plugging

SEGUNDO MANZANO: Atenção na construção da lógica em blocos

Diferente das diagramações clássicas, que não fornecem grandes subsídios para análise, os diagramas de blocos são realmente o melhor instrumento para avaliação do problema do fluxo de informações de um dado sistema. Por esse motivo deve-se resolver um problema de lógica (principalmente se for da área de processamento eletrônico de dados) usando um procedimento de desenvolvimento.

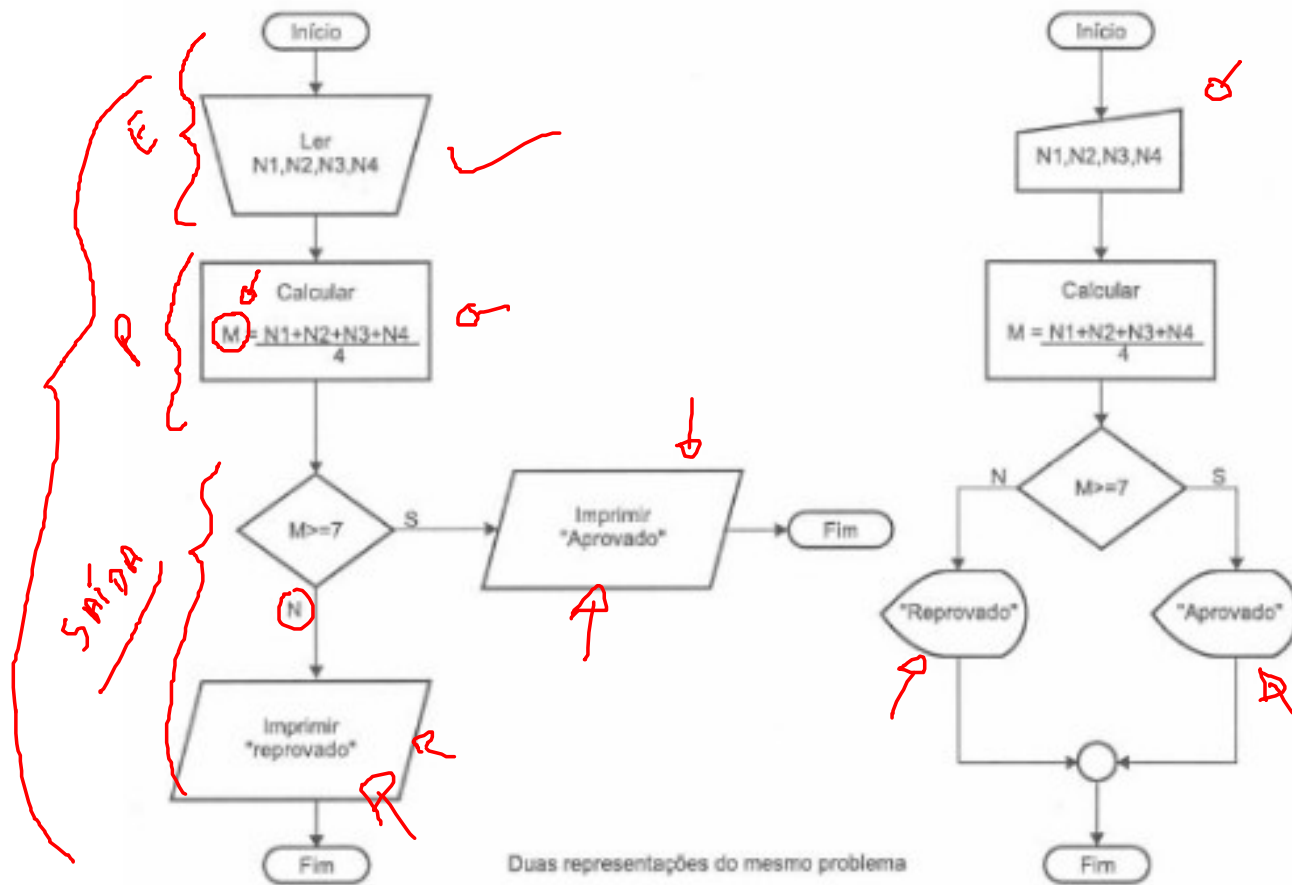
Para desenvolver um diagrama correto, deve-se considerar como procedimentos prioritários os itens seguintes:

- Os diagramas devem ser feitos e quebrados em vários níveis. Os primeiros devem conter apenas as ideias gerais, deixando para as etapas posteriores os detalhes necessários;
- Para o desenvolvimento correto de um fluxograma, sempre que possível, deve ser desenvolvido de cima para baixo e da esquerda para direita;
- É incorreto e “proibido” ocorrer cruzamento das linhas de fluxo de dados.



MODULAR





Conceitos

*Leia (A, B) → VARIÁVEIS
C ← A + B
Escreva C*

Antes de iniciar o estudo de programação, é necessário considerar que um **computador** nada mais é do que uma ferramenta utilizada para **solucionar problemas** que envolvam a manipulação de **informações**, sendo que essas informações classificam-se grosso modo em dois tipos básicos:

↓ INFORMAÇÃO
Dados

↓
Instruções

Conceitos

Os dados são representados pelas informações a serem tratadas (processadas) por um computador.

As instruções são representadas pelo conjunto de palavras-chaves (vocabulário) de uma linguagem de programação (Pascal, C, Basic, SmallTalk, Fortran, Cobol, Java) que tem por finalidade comandar, em um computador, o seu funcionamento e a forma como os dados armazenados devem ser tratados.

Variáveis e Tipos Primitivos

comuns / ESCALARES

Conceitos

Tem-se como definição de **variável** tudo aquilo que é sujeito a variações, que é incerto, instável ou inconstante.

E quando se fala de **computadores**, temos que ter em mente que o volume de informações a serem tratadas é **grande** e **diversificado**.

Todo dado a ser armazenado na **memória de um computador** deve ser **previamente identificado**, ou seja, primeiro é necessário saber qual o seu **tipo** para depois fazer o seu **armazenamento** adequado.

Identificadores

Um **identificador** é o nome pelo qual um componente será reconhecido.

Em um algoritmo, **identificadores** são usados para dar nome a **variáveis**, **constantes** e nomes de **rotinas**. módulo

Cada linguagem tem suas próprias regras para nomear seus **identificadores**, mas algumas delas estão presentes na maioria dos casos.

Regras para Identificadores

Um **identificador** deve conter apenas letras, números ou underline.

O primeiro caractere deve ser uma letra.

O nome de um **identificador** não pode conter espaços em branco. Conforme a necessidade, podemos substituí-los por um símbolo de underline.

Não são permitidos caracteres acentuados nem **símbolos** especiais como \$ # *.

Um **identificador** não pode ser uma palavra reservada da linguagem em questão.

NÃO POSSO
UTILIZAR
SÍMBOLOS
ESPECIAIS

Válido ou Inválido?

NOMEDOUSUARIO	Válido
25_delta	Inválido
telefone#	Inválido
delta_25	Válido
escreva	Inválido
z1	Válido
NOME DO USUARIO	Inválido



Variáveis

A **memória principal** de um computador é composta por células que podem armazenar dados.

Cada espaço de **memória** pode ser **alocado** para armazenar um determinado valor.

ap702





Variáveis

A **memória principal** de um computador é composta por células que podem armazenar dados.

Cada espaço de **memória** pode ser **alocado** para armazenar um determinado valor.

num



1

Tipos Primitivos

DINAMICAMENTE
TIPOADAS

FORTEMENTE
TIPOADAS

Cada **variável simples** deve ter um **tipo primitivo** relacionado a ela. Ele determina que tipo de valores vão poder ser colocados dentro da variável.

Existem basicamente **quatro famílias** de tipos primitivos. Cada qual serve para guardar dados de um determinado formato.

Cada **linguagem de programação** vai tratar seus tipos primitivos de uma maneira peculiar, com seus próprios intervalos.

Tipos Primitivos

NumeroJogador ← 21 ✓



PesoBalanca ← 56.8 ✓



PontoTuristico ← "Cristo" *simples*



ObraEncerrada ← falso



Tipos Primitivos

NumeroJogador ← 21



Inteiro

A: INT

1
2
3.5

PesoBalanca ← 56.8



Real

PontoTuristico ← "Cristo"



Literal

A: REAL
1.0
1.5

CHARACTER

ObraEncerrada ← falso



Lógico

Declaração de Variáveis

var

identificador : tipo primitivo

idade, totFilhos : Inteiro

peso, salario : Real

nome, endereço : Caractere / DIGITAL

casado, obraTerminada : Logico

True / False
VERDADEIRO / FALSO

Uso de Constantes

Se **variáveis** são espaços reservados cujo valor **varia**, as **constantes** são espaços ou valores cujo conteúdo **não variam**.

Tanto **variáveis** como **constantes** poderão ser utilizadas na elaboração de cálculos matemáticos.

variável composta?

vetores
matrizes
ligação

Classificação

Um **operador** se classifica de acordo com o **número de operandos** e a sua **finalidade**.

unário → **-8**

binário → **7-4**

Atribuição

OPERADORES

Atribui o **resultado** da expressão que está do lado **direito** à variável que está do lado **esquerdo**.

MANZANO

x $\leftarrow$ 4 + 5

leia "recebe"

m $:=$ (n1 + n2) / 2

variável

x = A = B = 5

B 5 A 5 x 5

Operadores Aritméticos

Usados na elaboração de **cálculos matemáticos**.

$$\begin{array}{r} 5 \overline{) 2,5} \\ \underline{10} \\ 0 \end{array}$$

$$\begin{array}{r} 5 \overline{) 2} \\ \underline{10} \\ 0 \end{array}$$

precedência ↓	-	subtração (unária)	$\ominus 5$	
	+	adição (unária)	$\oplus 5$	
	^	exponenciação	$5 \wedge 2$	5^2
	Div	<u>divisão inteira</u>	$5 \text{ Div } 2$	$= 2$
	Mod	resto da divisão %	$5 \text{ Mod } 2$	$= 1$ $5 \% 2$
	/	divisão	$5/2$	$= 2,5$
	*	multiplicação	$5*2$	$= 10$
	-	subtração (binária)	$5-2$	$= 3$
	+	adição (binária)	$5+2$	$= 7$

$$x \leftarrow 5 + 2^1 / 2 \rightarrow 6$$

$$x \leftarrow (5 + 2) / 2 \rightarrow 3,5,$$

$$x \leftarrow \underline{5 * 3} \text{ Mod } 2 \rightarrow 1$$

$$\begin{array}{r} 15 \overline{) 27} \\ \underline{14} \\ 13 \end{array}$$

$$20 \text{ Mod } 3 = 2$$

$$\begin{array}{r} 20 \overline{) 63} \\ \underline{40} \\ 23 \end{array}$$

Radiciação

$$\sqrt{25} \quad r \leftarrow \underline{25} \wedge \underline{(1/2)}$$

$$\sqrt[3]{27} \quad r \leftarrow \underline{27} \wedge \underline{(1/3)}$$

Operadores Relacionais (comparação)

Usados para fazer **comparações escalares**. Resulta sempre em um valor **lógico**.

$\rightarrow$	$\underline{=}$	igual a	5=3	}	F
$\rightarrow$	$\neq$	diferente de	8<>2		V
$\rightarrow$	$>$	maior que	4>7		F
$\rightarrow$	$<$	Menor que	2<6		V
$\rightarrow$	$\geq$	maior ou igual a	4>=4		V
$\rightarrow$	$\leq$	menor ou igual a	9<=7		F

Operadores Lógicos

Usados para fazer comparações entre lógicos. Resulta sempre em um valor lógico.

Proridade
NÃO
e
ou
Xou

e conjunção

$$(3 < 2) \text{ e } (9 > 7) = F$$

ou disjunção

$$(6 < > 3) \text{ ou } (1 > 7) = V$$

xou disjunção exclusiva

$$(2 < 5) \text{ xou } (1 < > 8) = F$$

nao negação (unário)

$$\text{nao } (5 = 6) = V$$

Tabela Verdade

p	q	$p \text{ e } q$
V	V	V
<u>V</u>	<u>F</u>	F
<u>F</u>	<u>V</u>	F
F	F	F

p	q	$p \text{ ou } q$
V	V	V
<u>V</u>	F	V
F	<u>V</u>	V
F	F	F

p	q	$p \text{ x ou } q$
V	V	F
V	F	V
F	V	V
<u>F</u>	<u>F</u>	F

p	não p
V	F
F	V

Quando 2 condições
forem iguais
prevalecerá o
Falso!

– Assinale a alternativa que contém a descrição correta de algoritmo.

- a) Algoritmo é uma coleção de livros de uma mesma matéria, normalmente relacionada à Engenharia de Software.
- b) Algoritmo é uma operação matemática usada, por exemplo, para calcular a intensidade sonora medida em decibéis.
- ~~c)~~ Algoritmo é uma descrição das etapas de resolução de um problema ou a indicação ordenada de uma seqüência de ações bem-definidas.
- d) Algoritmo é uma definição formal da hierarquia de funcionários de uma empresa de desenvolvimento de software de grande porte.

– Analise a figura a seguir e assinale a alternativa correta.



Essa figura é um exemplo de

- a) Portugol.
- b) Pseudocódigo.
- ~~c) Diagrama de Chapin.~~
- d) Português estruturado.

→ GRÁFICO

– Fluxograma é um tipo de

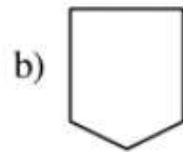
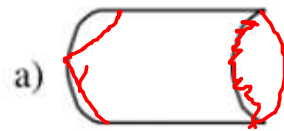
- ~~a)~~ representação gráfica de algoritmos.
- b) informação sobre tipos de dados.
- c) livro de análise de sistemas.
- d) análise de sistemas.

- Assinale a alternativa que corresponde à simbologia abaixo, utilizada na área de tecnologia da informação.

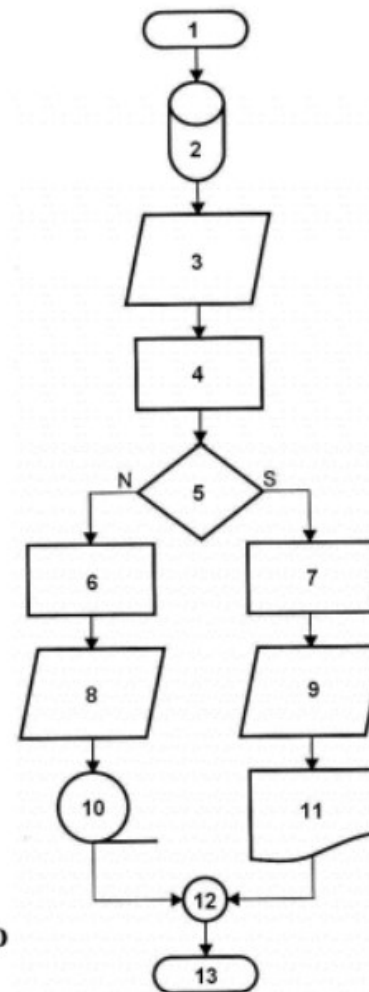


- a) Controlador
- b) Intercalação
- ~~c) Módulo de subrotina~~
- d) Checagem de operação auxiliar

– Assinale a alternativa que contém o símbolo utilizado em fluxogramas de sistemas de processamento de dados para indicar saídas impressas como, por exemplo, a emissão de um ou vários relatórios.



– Observe o gráfico abaixo e responda qual o significado do símbolo representado pelo número 5.



- a) Terminal
- b) Conector
- ~~c) Decisão~~
- d) Saída de dados em vídeo

– Relacione os símbolos utilizados no fluxograma com seus respectivos nomes e, a seguir, assinale a alternativa que contém a sequência correta.

1 - Terminal (3) 

2 – Seta de fluxo de dados (1) 

3 - Processamento (6) 

4 – Entrada de dados ou operação manual (4) 

5 – Entrada e saída de dados (7) 

6 – Saída de dados em vídeo (5) 

7 – Saída de dados em impressora (8) 

8 - Decisão (2) 

- ~~a)~~ 3 – 1 – 6 – 4 – 7 – 5 – 8 – 2
b) 3 – 1 – 6 – 5 – 7 – 4 – 8 – 2
c) 1 – 3 – 4 – 6 – 5 – 7 – 2 – 8
d) 1 – 3 – 6 – 4 – 7 – 5 – 8 – 2

___ – Assinale a alternativa que contém uma expressão lógica com resultado VERDADE. Considere $X = 7$ e $Y = 4$.

- a) $(X > 5) .E. (.NÃO. (Y < 3)) = V$
- b) $(X = 5) .OU. (Y > 8) = F$
- c) $(Y > 10) .E. (X = 7) = F$
- d) $.NÃO. (X = 7) = F$

– Marque 1 para dados do tipo inteiro e 2 para dados de outros tipos.

(2) "582.4" → Literal

(2) .verdadeiro. Lógico

(1) 105

(1) -102

(2) "0" → Literal

(2) "informação" → Literal

(2) 0.82 → Real

a) 2, 2, 1, 1, 1, 2, 1

~~b) 2, 2, 1, 1, 2, 2, 2~~

c) 1, 1, 1, 2, 2, 1, 2

d) 1, 2, 2, 2, 1, 1, 1

– Assinale a alternativa que contém somente exemplos de operadores lógicos.

- ~~a)~~ .e. .ou. .não.
- b) .e. .para. .ou.
- c) .não. .sim. .talvez.
- d) .enquanto. .ou. .talvez.

– Assinale a alternativa que contenha somente nomes válidos de variáveis.

- a) ~~índice~~, ~~#pagina~~, contexto
- ~~b)~~ nome1, sobrenome2, senha3
- c) 2-~~nome~~, sobrenome, endereço
- d) 1-~~nome~~, 2-sob~~renome~~, 3-sen~~ha~~

– Qual operador lógico é utilizado para estabelecer a inversão do resultado lógico a uma determinada condição?

- ~~a) .não.~~
- b) .ou.
- c) .e.
- d) .and.

– Assinale a alternativa que completa corretamente a lacuna da afirmativa a seguir.

São caracterizados como tipos _____ os dados numéricos positivos ou negativos, excluindo-se destes qualquer fracionário.

- a) caracteres
- b) lógicos
- ~~c) inteiros~~
- d) reais

– Marque a alternativa que identifica o tipo de operador lógico utilizado na tabela abaixo.

Condição 1	Condição 2	Resultado
Falsa	Falsa	Falso
Verdadeira	Falsa	Verdadeiro
Falsa	Verdadeira	Verdadeiro
Verdadeira	Verdadeira	Verdadeiro

- a) .e.
- ~~b) .ou.~~
- c) .if.
- d) .não.