

Aplicações Informáticas



Introdução à Programação

Linguagem

- **Conceito de Linguagem**

- O termo linguagem designa um sistema organizado de símbolos, complexo, extenso e com propriedades particulares que desempenha uma função de codificação, estruturação e consolidação dos dados sensoriais, transmitindo-lhe um determinado sentido ou significado e permitindo ao homem comunicar as suas experiências e transmitir os seus saberes. É, portanto, um sistema de troca de informações. Além desta função de comunicação, a linguagem desempenha ainda outras funções, entre as quais a apelativa, expressiva, descritiva, estética, argumentativa e persuasiva).



Linguagens

Linguagens naturais são as linguagens que utilizamos para comunicar no dia a dia, aquela que encontramos nos livros, revistas, nos jornais e que as pessoas falam, como o português, o inglês e o espanhol. Elas não foram projectadas pelas pessoas (muito embora as pessoas tentem colocar alguma ordem nelas); elas evoluíram



Linguagens formais são linguagens que foram projectadas por pessoas, para aplicações específicas. Por exemplo, a notação que os matemáticos usam é uma linguagem formal, que é particularmente boa em denotar relações entre números e símbolos. Os químicos usam uma linguagem formal para representar a estrutura química das moléculas. E, mais importante:

Linguagens de programação são linguagens formais que foram desenvolvidas para expressar computações.



Linguagens

- Comparações entre estas duas linguagens são inevitáveis.
- Por um lado, o funcionamento da linguagem natural é algo mais complexo que o das linguagens formais. A linguagem natural é construída historicamente, tem uma estrutura própria e é dinâmica. Além disso, ela é social, ou seja não pertence a ninguém. O significado dos seus termos varia de acordo com o seu contexto. Em comparação com as linguagens formais, a linguagem natural tem um poder de expressividade maior. Ela é mais flexível, contudo mais ambígua.
- Por outro lado, a estrutura de uma linguagem formal é mais simples. A sua gramática possui normas reduzidas, o significado dos seus termos é fixo e a ambiguidade é nula. Contudo, o poder expressivo é menor.

Algoritmo



Um algoritmo pode ser definido como uma sequência ordenada, sem ambiguidade, de passos que levam à solução de um dado problema, utilizando um número finito de

Apesar de o termo ser novo em si, o conceito é bastante familiar. As indicações dadas para chegar até uma determinada rua, constituem um algoritmo para encontrar essa rua. Uma receita de cozinha é uma forma muito familiar de algoritmo.

Algoritmo

Vindo da Maia e Matosinhos pela estrada da Circunvalação (EN12)

Deverá seguir pela EN12 até encontrar o IPO à sua direita. Coloque-se à direita nos semáforos para virar à direita para a Rua António Bernardino Almeida (contornando o IPO) e siga em frente, mantendo-se na esquerda até ao cruzamento com a Rua do Dr. Plácido da Costa, onde deverá virar à esquerda. Mantenha-se na direita até ao cruzamento com a Rua Dr. Roberto Frias onde deverá virar à direita. Mantenha-se na esquerda até ao cruzamento com a Via



Algoritmo



2 emb. bolacha-maria
4 gema(s) de ovo
125 g manteiga
250 g açúcar
1 caneca café muito forte
côco ralado
bolocho-maria ralada

1. Bater as gemas, o açúcar e a manteiga muito bem até a mistura ficar cremosa.
2. Fazer café forte e passe as bolachas, uma a uma pelo café, não muito quente, e deixam-se arrefecer.
3. Coloca-se uma primeira fiada de bolachas num prato, como por exemplo em forma de flor, e depois barra-se com o creme.
4. Depois volta-se a colocar outra fiada de bolachas embebidas em café e depois o creme, e assim sucessivamente.
5. Por fim barra-se o bolo por fora e polvilha-se com o coco ralado e bolacha ralada.

Algoritmo

Algoritmo é uma sequência finita de passos lógicos necessários para realizar uma determinada tarefa. Uma simples acção como escovar os dentes pode ser descrita, de modo a formar o que chamamos de Algoritmo.

1. Pegar na escova de dentes
2. Pegar na pasta de dentes, com a outra mão
3. Colocar a pasta de dentes na escova
4. Colocar a escova de dentes na boca
5. Escovar os dentes
6. Retirar a escova da boca

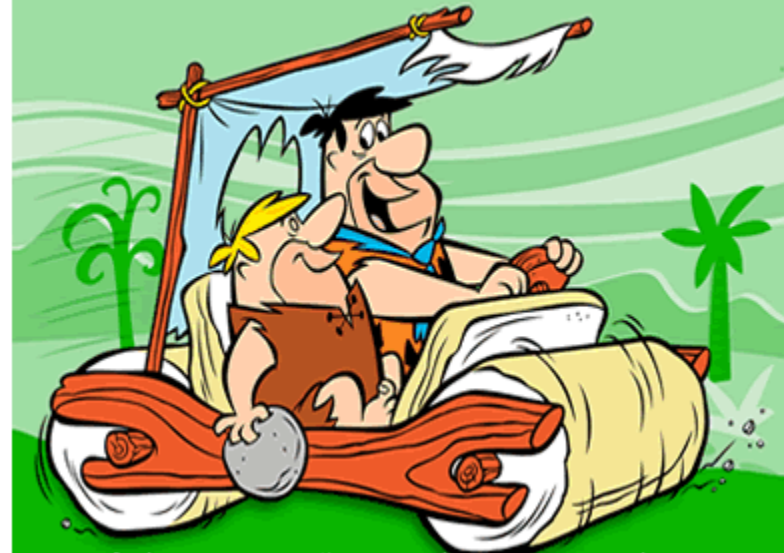
Faça o mesmo para :

- A substituição de uma lâmpada



-A substituição de um pneu

Pneus Flinstones



Os únicos que não precisam ser calibrados!

Resolução Algoritmo



Substituição de uma lâmpada

1. [Preparar acessos à lâmpada fundida]

1.1 Repetir enquanto não chegar à escada

1.1.1 Dar passos

1.2 Pegar objecto (escada)

1.3 Repetir enquanto não chegar debaixo lâmpada fundida

1.3.1 Dar passos

2. [Retirar lâmpada fundida]

2.1 Repetir enquanto não chegar à lâmpada fundida

2.1.1 Subir degraus

2.2 Repetir enquanto não soltar a lâmpada fundida

2.2.1 Rodar objecto (L.F.) no sentido indirecto

2.3 Repetir enquanto não chegar ao chão

2.3.1 Descer degraus

3.[Escolher lâmpada nova]

3.1 Repetir enquanto não chegar junto gaveta das lâmpadas novas

3.1.1 Dar passos

3.2 Repetir enquanto houver lâmpadas novas ou potência L.N. diferente de potência L.F.

3.2.1 Pegar objecto (L.N.)

3.2.2 Se potência L.N. = Potência L.F.

3.2.2.1 Então largar a L.F.

3.2.2.2 Senão largar a L.N.

4. [Colocação da lâmpada nova]

4.1 Repetir enquanto não chegar à escada

4.1.1 Dar passos

4.2 Repetir enquanto não chegar ao casquilho

4.2.1 Subir degraus

4.3 Repetir enquanto não firmar a lâmpada nova

4.3.1 Rodar objecto (L.N.) no sentido directo

4.4 Repetir enquanto não chegar ao chão

4.4.1 Descer degraus

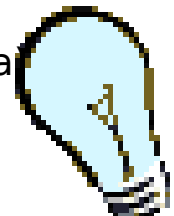
5. [Arrumar escada]

5.1 Pegar objecto (escada)

5.2 Repetir enquanto não chegar ao sítio da escada

5.2.1 Dar passos

5.3 Largar objecto (escada)





Resolução Algoritmo

Substituição de um pneu

1. [Buscar macaco, chave e pneu sobresselente]
 - 1.1 Repetir até chegar ao carro
 - 1.1.1 Dar passos
 - 1.2 Abrir a mala
 - 1.3 Pegar macaco, chave e pneu bom
 - 1.4 Repetir até chegar ao pneu furado
 - 1.4.1 Dar passos
2. [Colocar o macaco em posição correcta e elevar o carro]
 - 2.1 Largar o pneu e a chave
 - 2.2 Encaixar macaco
 - 2.2.1 Deslocar o macaco no sentido do carro até encaixar
 - 2.3 Elevar o carro
 - 2.3.1 Repetir até carro elevado
 - 2.3.1.1 Rodar manivela no sentido horário
3. [Retirar o pneu furado e substituí-lo]
 - 3.1 Pegar na chave
 - 3.2 Repetir quatro vezes
 - 3.2.1 Deslocar a chave até encaixar na porca
 - 3.2.2 Repetir até porca cair
 - 3.2.2.1 Rodar sentido anti-horário
 - 3.3 Largar chave



- 3.4 Pegar no pneu furado
- 3.5 Deslocá-lo sentido contrário ao carro
- 3.6 Largar pneu furado
- 3.7 Pegar no pneu bom
- 3.8 Deslocar pneu no sentido do carro
- 3.9 Encaixar pneu
- 3.10 Pegar na chave
- 3.11 Repetir quatro vezes
 - 3.11.1 Pegar na porca
 - 3.11.2 Encaixar porca
 - 3.11.3 Pegar chave
 - 3.11.4 Repetir até ficar apertado
 - 3.11.4.1 Rodar no sentido horário
- 3.12 Largar chave
4. [Guardar o macaco, chave e o pneu furado]
 - 4.1 Baixar o carro
 - 4.1.1 Rodar manivela no sentido contrário ao horário
 - 4.2 Desencaixar macaco
 - 4.3 Pegar pneu furado+macaco+chave
 - 4.4 Repetir até à mala
 - 4.4.1 Dar passos
 - 4.5 Largar chave+macaco+pneu furado
 - 4.6 Fechar mala



Dados e tipos de dados

A resolução de problemas através de algoritmos requer a utilização de dados.

Estes podem variar de tipo. Por exemplo, no cálculo da média das notas dos alunos desta turma, o nome do aluno é alfabético, enquanto as suas notas são do tipo numérico. Assim podemos considerar os seguintes tipos de dados:

Estrutura de dados

Primitivas

Númericos

Inteiros

13, -6, 208, -298

S

Reais

23.8, -8.910

S

Alfanuméricos

'Hoje',
'Ab*12'

Lógicos

Verdadeiro,
Falso

Não Primitivas

Vectores

Matrizes



Operadores



Aritméticos

| | |
|---|---------------|
| + | Adição |
| - | Subtracção |
| * | Multiplicação |
| / | Divisão |
| ^ | Exponenciação |

Relacionais

| | |
|-----|-------------------------|
| = | Igualdade |
| < | Menor que |
| > | Maior que |
| < > | Diferente |
| < = | Menor do que ou igual a |
| = > | Maior do que ou igual a |

Lógicos

| | |
|---------|-----------|
| AN D | Conjunção |
| OR | Disjunção |
| NO T | Negação |

Operadores Aritméticos

Prioridade

| Ordem | Operador | Significado |
|-----------|----------|--|
| 1 | () | Parênteses |
| 2 | \wedge | Exponenciação (direita para a esquerda) |
| 3 | - , + | Sinal dos operandos (menos e mais) (direita para a esquerda) |
| 4 | * , / | Multiplicação e divisão (esquerda para a direita) |
| Exemplos: | + , - | Adição e subtração (esquerda para a direita) |

$3+6*5$

9

4
5



$3+6*5$

30

33



$8+7*3+4*5$

2

2

1

0

29

49



$(8+7)*(3+4)*5$

5

15

7

10

5

525



$2^3 = 2^3^2$

9

512



Variáveis



Variável: na Matemática, é uma entidade capaz de representar um valor ou expressão;

em Programação, é um objecto que representa uma expressão.

Como na matemática, na algoritmia (programação) o uso de variáveis permite a especificação de uma fórmula geral de cálculo.

As variáveis também têm nomes e recebem valores. Uma variável pode receber muitos valores diferentes, mas num dado instante, ela tem

Regras para dar um nome a uma variável

- ✓ O nome da variável começa sempre por uma letra;
- ✓ Os outros caracteres podem ser letras, dígitos numéricos e alguns caracteres especiais;
- ✓ Espaços em branco não são permitidos dentro do nome da variável;
- ✓ Os nomes devem ser mnemónicos e altamente sugestivos.

Total



A123



3Alt



A■B



Lado2



Var_Com
p



X+Y



Z/n





Atribuição

A operação de atribuição é uma forma de especificar que a uma variável será dado um valor.

A operação de atribuição será indicada pelo símbolo \leftarrow .

Por exemplo, o comando $A \leftarrow 3$, indica que à variável A é atribuído o valor 3.

Na seguinte sequência de operações:

$A \leftarrow 16$

$A \leftarrow -27$

$A \leftarrow 1$

O valor da variável A após as três operações seria 1. Os valores 16 e -27 foram

Até aqui utilizamos a operação de atribuição de uma constante a uma variável. Mas o lado direito do comando de atribuição pode ser qualquer expressão, onde a expressão é uma combinação de variáveis, constantes e operadores.

O resultado dessa expressão é atribuído à variável indicada.

Exemplo:

$A \leftarrow 4$

$B \leftarrow A + 3 * 2$ (Expressão)

Compatibilidade de Tipos

Todas as variáveis pertencem a um tipo de dados e podem receber apenas valores (dados) daquele tipo.

Se tentarmos atribuir um valor a uma variável que não seja do mesmo tipo, o que acontece?

Variáveis:

| | |
|---|--------------|
| A | Inteira |
| B | Real |
| C | Alfanumérica |

C
U
I
D
A
D
O

Automaticamente

A ← 31.9
O valor é truncado
A ← 31

B ← 86
O valor é convertido
B ← 86.0

A ← 20
B ← -2.5
C ← 'variável'



A ← 'variável'
B ← 'variável'
C ← 20



Não existe compatibilidade



Operadores Lógicos

AND (Conjunção)

| C1 | C2 | C1 AND C2 |
|------------|------------|------------|
| Verdadeiro | Verdadeiro | Verdadeiro |
| Verdadeiro | Falso | Falso |
| Falso | Verdadeiro | Falso |
| Falso | Falso | Falso |

OR (Disjunção)

| C1 | C2 | C1 OR C2 |
|------------|------------|------------|
| Verdadeiro | Verdadeiro | Verdadeiro |
| Verdadeiro | Falso | Verdadeiro |
| Falso | Verdadeiro | Verdadeiro |
| Falso | Falso | Falso |

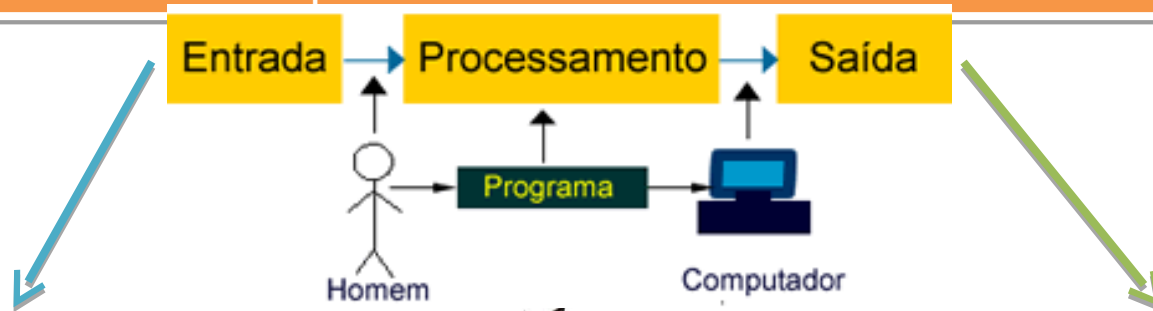


NOT (Negação)

| C1 | NOT C1 |
|------------|------------|
| Verdadeiro | Falso |
| Falso | Verdadeiro |

Instruções de Entrada e Saída

Um algoritmo, pouco valor terá , se não puder recolher dados para posteriormente serem processados e de seguida ser apresentado o resultado para o nosso problema.



A instrução de leitura:

READ (variável,
variável, ... , variável)

Exemplo:

READ (A, B, C)



A instrução de escrita:

PRINT ()

Exemplo:

PRINT (3, 5)
PRINT (A, B, C)
PRINT ('Olá')

Algoritmo

Diferentes abordagens para a construção de um algoritmo

Descrição Narrativa

Este método consiste na descrição dos passos utilizando a linguagem natural.

Preparação

Crepes

Misture a farinha com o açúcar e junte o leite com o ovo previamente batido. Adicione a margarina derretida e tempere com 1 pitada de sal. Deixe repousar cerca de 30 minutos para que a massa não fique elástica.

Unte 1 frigideira anti-aderente com margarina e aqueça, quando estiver bem quente, deite um pouco de massa na frigideira e faça-a rodar de forma a cobrir o fundo, deixe cozer até ficar louro e vire para cozer do outro lado. Retire o crepe para um prato e repita as operações até acabar a massa.

Molho

Misture o chocolate em pó com o açúcar e dissolva-o com o café, acabado de fazer. Junte a margarina e mexa até todos os ingredientes estarem bem misturados.

Disponha 1 bola de gelado em cada crepe e dobre o crepe ao meio. Regue com o chocolate quente.



A imprecisão da linguagem natural, leva ao perigo de má interpretação ou perda de informação e detalhes confusos podem levar a erros. Por estas razões, a linguagem natural não é adequada como o único veículo para a expressão de algoritmos.

Algoritmo

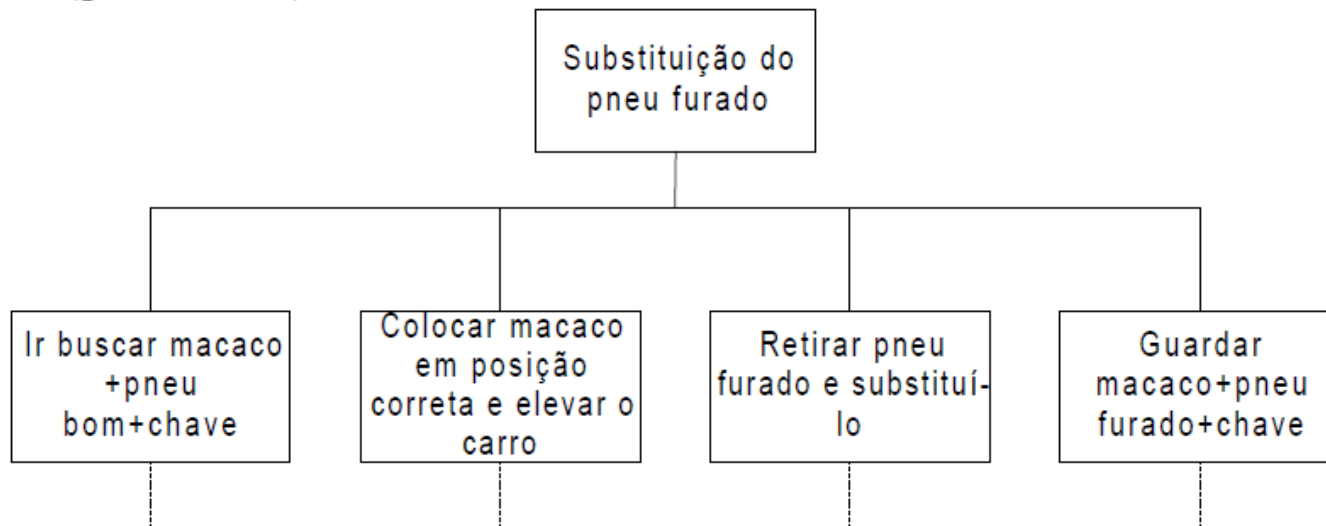
Diferentes abordagens para a construção de um algoritmo

Top - Down

Esta técnica consiste num paradigma de atacar o problema de forma gradual. Após o entendimento completo do problema, basta dividi-lo em partes menores para que os detalhes sejam tratados individualmente e a solução como um todo seja alcançada com a união das resoluções de todas as partes.

Assim, encarando o problema de uma especificação geral até suas minúcias, a resolução caminha de maneira natural do topo (mais abrangente - top) até o baixo nível (mais preciso - down). Sem delongas, o método top-down permite estruturar o pensamento, pois a resolução caminha de maneira natural do topo (mais abrangente - top) até o baixo nível (mais preciso - down).

TopDown (graficamente):
a



Algoritmo

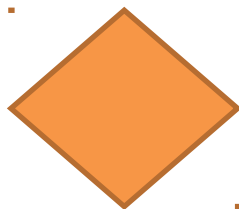
Diferentes abordagens para a construção de um algoritmo

Fluxograma (ou Diagrama de Fluxo)

Esta é uma representação gráfica que emprega formas geométricas padronizadas para indicar as diversas acções e decisões que devem ser executadas para resolver um problema.

Início e Fim

E/S de
Dados

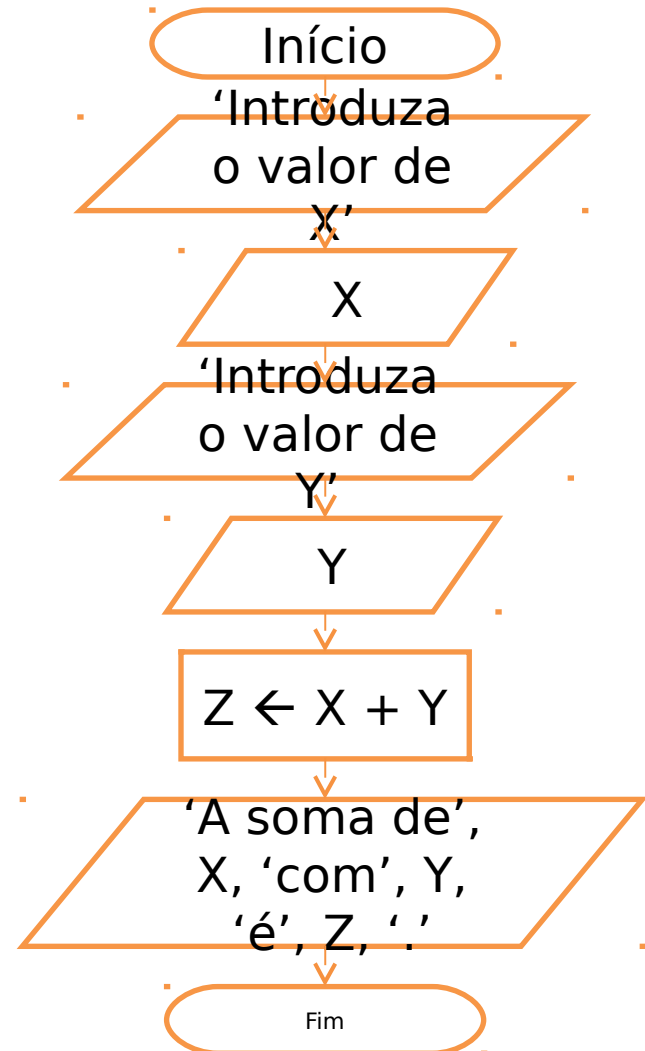


Decisão



Processo

Soma de dois números reais, lidos a partir do teclado



Algoritmo

Diferentes abordagens para a construção de um algoritmo

Pseudocódigo (PSEUDO-CÓDIGO)

Esta abordagem aspira a retirar ambiguidades e indefinições da linguagem natural, apresentando restrições para tal, utilizando uma forma estruturada.

O resultado é um tipo de “linguagem simplificada de programação”, similar em sabor a várias linguagens de programação.

ALGORITMO SOMA 1.0

Este algoritmo lê dois valores a partir do teclado e determina a sua soma, imprimindo-a.

```
S1 [Leitura do primeiro valor]
    READ ( X )
S2 [Leitura do segundo valor]
    READ ( Y )
S3 [Cálculo da soma dos dois valores]
    Z ← X + Y
S4 [Impressão do resultado]
    PRINT ( ' A soma de', X, 'com', Y, 'é', Z)
S5 [Termina]
    EXIT
```





'seudocódigo

Todo o algoritmo deve ter um nome, em maiúsculas, precedido pela palavra **ALGORITMO**.

ALGORITMO SOMA 1.0

Este algoritmo lê dois valores a partir do teclado e determina a sua soma, imprimindo-a.

```
S1 [Leitura do primeiro valor]
  READ ( X )
S2 [Leitura do segundo valor]
  READ ( Y )
S3 [Cálculo da soma dos dois valores]
  Z ← X + Y
S4 [Impressão do resultado]
  PRINT ( ' A soma de', X, 'com', Y, 'é', Z)
S5 [Termina]
  EXIT
```

Fim lógico
do
algoritmo.

O nome do algoritmo é seguido de uma **descrição** breve, clara, precisa e concisa do mesmo.

Descrição resumida do passo em particular. Seguido da descrição, está um comando ou uma série de comandos.

A primeira letra do nome do algoritmo, seguida de um número, identifica os passos do algoritmo.

Fim físico
do
algoritmo.

Pseudocódigo



Lista de Variáveis do ALGORITMO SOMA 1.0

| Nome | Tipo | Gama | Significado |
|------|---------|----------------------|-----------------------------------|
| X | Inteira | $[-\infty, +\infty]$ | O primeiro valor |
| Y | Inteira | $[-\infty, +\infty]$ | O segundo valor |
| Z | Inteira | $[-\infty, +\infty]$ | Valor da soma das variáveis X e Y |

Algumas Funções

| Funções | | |
|----------------|-------------------------|----------------------------------|
| Nome da Função | Argumento e tipo | Significado |
| ABS () | Real / Inteira | Valor sem sinal |
| SQRT () | Real / Inteira ≥ 0 | Raiz Quadrada |
| TRUNC () | Real | Apenas a parte inteira do número |
| ROUND () | Real | Valor arredondado |

Estruturas de Controlo

fluxo dos dados num algoritmo, existem estruturas base que condicionam o fluxo dos dados conforme as acções a realizar, permitindo tomar decisões no decorrer do processo.

Estruturas de controlo

É a mais elementar, pois as acções são executadas sequencialmente pela ordem em que ocorrem

Estrutura Sequencial

Estruturas de Selecção (Decisão)

Decide se uma acção ou grupo de acções deve ou não ser executado dependendo do valor de uma variável ou de uma expressão

Estruturas Repetitivas

Repete a execução de um grupo de acções



Estrutura Sequencial



E a mais elementar, pois as acções são executadas sequencialmente pela ordem em que ocorrem

ALGORITMO SOMA 1.0

Este algoritmo lê dois valores a partir do teclado e determina a sua soma, imprimindo-a.

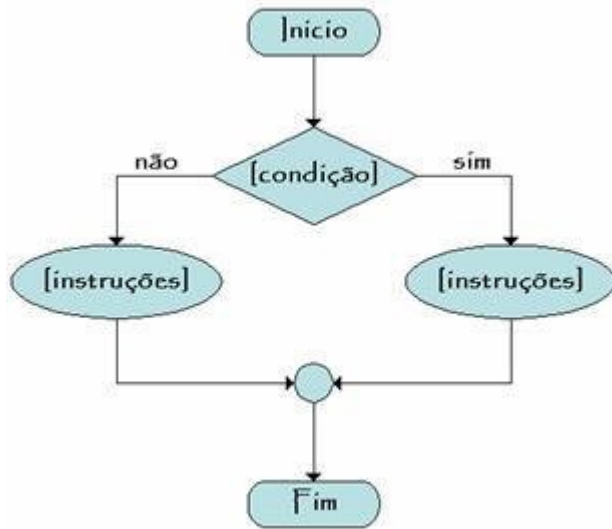
```
S1 [Leitura do primeiro valor]
    READ ( X )
S2 [Leitura do segundo valor]
    READ ( Y )
S3 [Cálculo da soma dos dois valores]
     $Z \leftarrow X + Y$ 
S4 [Impressão do resultado]
    PRINT ( ' A soma de', X, 'com', Y, 'é', Z)
S5 [Termina]
    EXIT
```



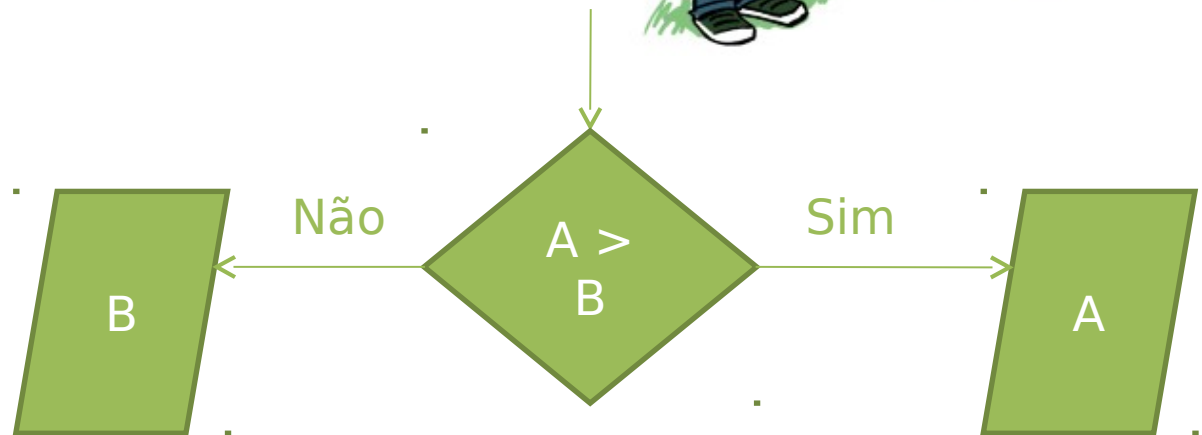
Estrutura de Seleção (Decisão)

IF (condição)
THEN (instruções)
ELSE (instruções)

Decide se uma acção ou grupo de acções deve ou não ser executado dependendo do valor de uma variável ou de uma expressão



IF $A > B$
THEN PRINT (A)
ELSE PRINT (B)



Estrutura de Seleção (Decisão)

ALGORITMO MAIOR 1.0

Este algoritmo lê dois valores a partir do teclado e determina o maior valor, imprimindo-o.

S1 [Leitura do primeiro valor]

 READ (X)

S2 [Leitura do segundo valor]

 READ (Y)

S3 [Cálculo do maior valor e determina o valor absoluto da diferença dos dois valores]

 IF $X > Y$

 THEN Diferença $\leftarrow X - Y$

 PRINT ('O 1º valor é superior ao 2º')

 ELSE Diferença $\leftarrow Y - X$

 PRINT ('O 2º valor é superior ao 1º')

S4 [Impressão do resultado]

 PRINT (' A diferença entre dois valores é ', Diferença)

S5 [Termina]

 EXIT



