

APROG – Algoritmia e Programação



Emanuel Cunha Silva

ecs@isep.ipp.pt

Apresentação

Enquadramento

Enquadra-se nas Ciências de Base da Engenharia Informática e visa dar conhecimentos estruturantes na área científica das Ciências da Computação.

Objetivos

- Introdução de conceitos fundamentais das Ciências da Computação associados à programação
- Conceção de algoritmos aplicando adequadas metodologias de programação
- Análise e resolução de problemas computacionalmente
- Codificação de algoritmos em linguagem Java (perspetiva procedimental)
- Assumir atitudes de aprendizagem ativa, colaborativa e responsável, de trabalho persistente, individual e em grupo, e de aplicação de espírito crítico na análise e resolução de problemas

Apresentação

Bibliografia

- Sítio da disciplina: moodle.isep.ipp.pt
- “Programação, algoritmos e estruturas de dados”; João Pedro Neto; Escolar Editora
- “Java: How to program (How to program series); Deitel & Deitel; Prentice Hall

Introdução

Informática

Informação + Automática

Computador

Máquina eletrónica-digital que executa algoritmos tendo em conta os dados de entrada a serem processados

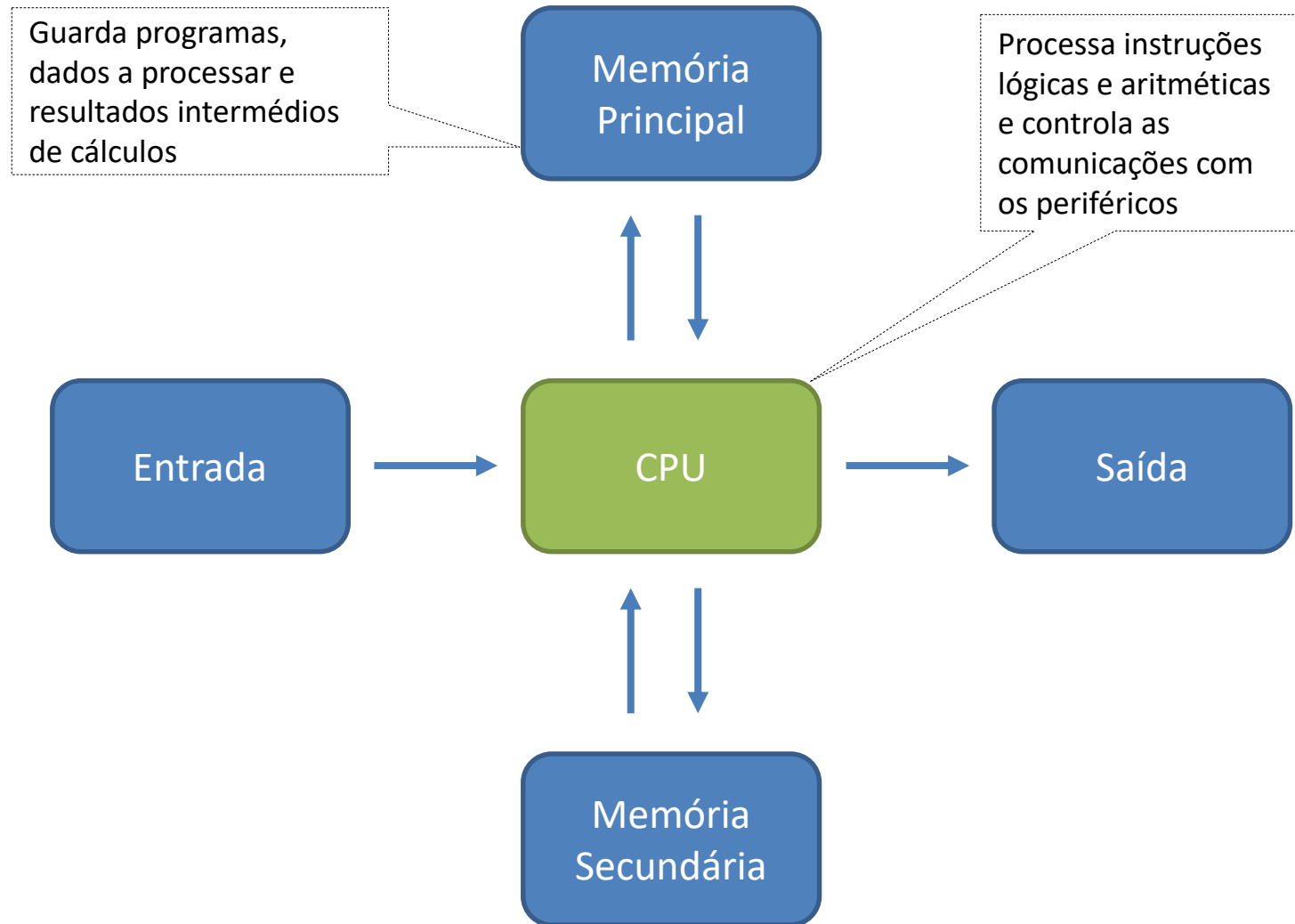
Hardware

Componente física do computador: componentes mecânicos, elétricos, eletrónicos, magnéticos.

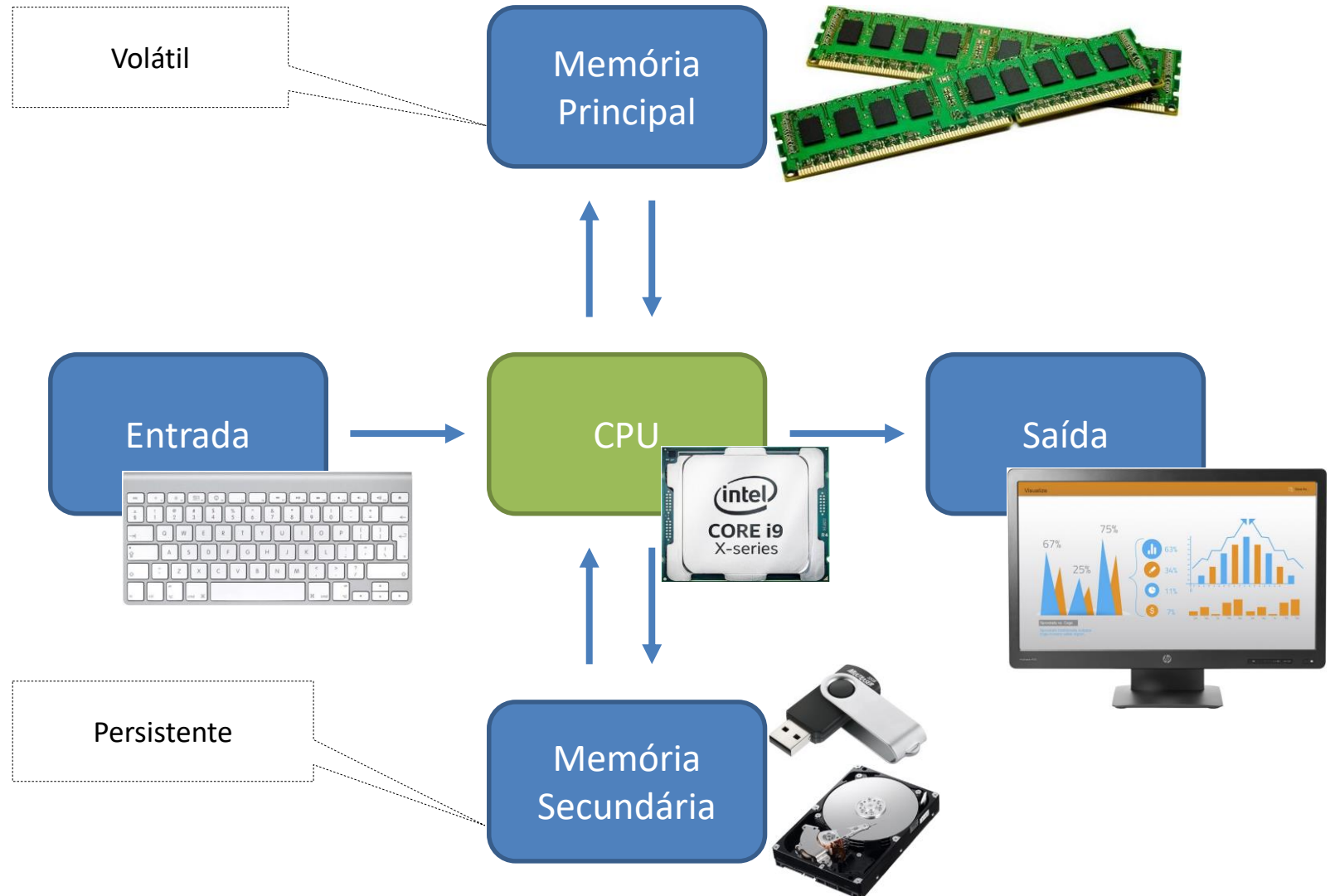
Software

Componente lógica do computador: Sistema operativo, ferramentas de desenvolvimento, aplicações,

Principais componentes de um computador



Principais componentes de um computador



Sistema operativo

- Permite a gestão dos recursos do computador: memória, periféricos
- Executa tarefas

Ficheiro

- Conjunto de dados agrupados sob um determinado nome que o identifica
- O SO é capaz de o reconhecer e manipular

Programa

- Ficheiro que descreve uma tarefa específica através de uma linguagem conhecida pelo SO.
- O SO é capaz de reconhecer um programa e executar a tarefa descrita no seu conteúdo

Programas

Programação estruturada

- Define um conjunto de regras para elaboração de programas
- Permite fasear o processo de construção de um programa descrevendo o processo computacional de um modo não ambíguo
- baseia-se no desenho modular dos programas e no refinamento gradual do topo para a base
- Segundo este paradigma: Programa = Estrutura de Dados + Algoritmo

Algoritmo

Sequência finita e não ambígua de instruções que descrevem os passos lógicos necessários à realização de uma dada tarefa ou resolução de um problema

Estruturas de dados

- Representam o modo como os dados são organizados, acedidos e alterados
- Tipos primitivos
 - Números Inteiros, números reais, caracteres, booleano (V ou F)
- Tipos não primitivos (complexos)
 - Tipos indexados mono e multidimensionais
 - Listas, filas
 - Árvores, grafos

Conceitos fundamentais de programação



A word cloud of programming concepts. The words are arranged in a dense, overlapping cluster. The colors of the words include shades of blue, green, yellow, orange, and red. The words are: TYPES, STRINGS, REAL, ARRAYS, OBJECTS, BOOLEAN, NUMBERS, DATA, INTEGERS, STRUCTURES, BINARY, VARIABLES, DATE, QUERIES, TIME, and LISTS.

TYPES
STRINGS
REAL
ARRAYS
OBJECTS
BOOLEAN
NUMBERS
DATA
INTEGERS
STRUCTURES
BINARY
VARIABLES
DATE
QUERIES
TIME
LISTS

Resolução de problemas

- Analisar o problema
 - Conhecer bem o problema
 - Descrever o problema: subdividir, detalhar
- Resolver o problema passo-a-passo
 - Verificar se não há ambiguidade na solução apresentada
- Implementar a solução
 - Numa linguagem de programação



Algoritmo

Um algoritmo é constituído por um conjunto de expressões simbólicas que representam ações (escolher, atribuir, etc.), testes de condições (estruturas condicionais) e estruturas de controlo (ciclos na estrutura sequencial do algoritmo) de modo a especificar o problema e respetiva solução

É representado através de uma linguagem com sintaxe e semântica associada

- Linguagem natural
 - português estruturado, inglês, etc.
- Pseudocódigo
 - linguagem intermédia. Aplica a linguagem natural em conjunto com construtores usados em linguagens de programação
 - mais precisa que a linguagem natural
- Notação gráfica
 - Fluxogramas

Representação de Algoritmos

Pseudocódigo

INÍCIO ou FIM

LER()

ESCREVER()

SE...ENTÃO...SENÃO

PARA...ATÉ...FAZER

ENQUANTO...FAZER

FAZER...ENQUANTO

PROCEDIMENTO/FUNÇÃO

Fluxograma



Início ou fim



Entrada de dados



Saída de resultados



Processamento



Tomada de decisão



módulo

Conceitos Fundamentais

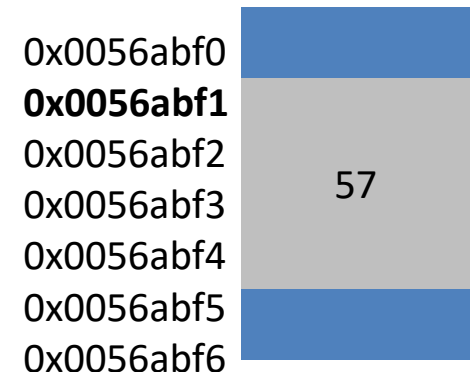
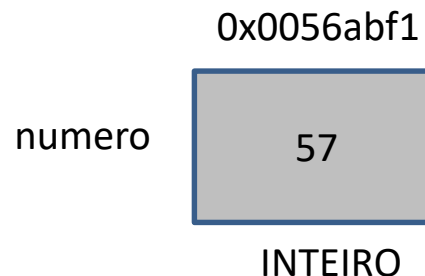
Variável

Unidade de memória de onde se pode aceder a uma informação de um dado tipo.

Durante a execução do algoritmo o valor de uma variável pode ser modificado

Quando uma variável é declarada e associada a um valor, 4 atributos fundamentais são considerados:

- Nome (deve ser sugestivo e curto)
- Tipo
- Valor (conteúdo)
- Endereço



Conceitos Fundamentais

Tipos de Dados

*Um **tipo** é um conjunto de valores relacionados por um conjunto de operações*

Tipos primitivos

- Números Inteiros (INTEIRO)
- Números reais (REAL)
- Carácter (CHARACTER)
- Cadeias de caracteres (TEXT)
- Booleano (BOOLEAN)

Tipos não primitivos (complexos)

- Tipos indexados mono e multidimensionais
 - referenciar e guardar valores de um tipo primitivo único.

Conceitos Fundamentais

Operadores

Aritméticos

- $*$, $/$, $^$: multiplicação, divisão real, potenciação
- DIV, MOD, % : divisão inteira, resto da divisão inteira, resto da divisão inteira
- $+$, $-$: soma, subtração

Lógicos

- E : conjunção
- OU : disjunção
- NÃO : negação

Relacionais

- $<$, $<=$: menor, menor ou igual
- $=$, \neq : igual, diferente
- $>$, $>=$: maior, maior ou igual

Atribuição

- \leftarrow

Conceitos Fundamentais

Estrutura do algoritmo

- (1) Declaração das variáveis necessárias
- (2) Leitura dos dados
- (3) Processamento ...
- (4) Escrita dos resultados

INICIO

ED: (1)

variavel1, variavel2 INTEIRO

variavel3, variavel4 REAL

LER (variavel1, variavel2) (2)

variavel3 \leftarrow variavel1 + variavel2 (3)

ESCREVER (variavel3) (4)

FIM

Conceitos Fundamentais

Estrutura de controlo do algoritmo

- Sequência
 - Executar instruções sequencialmente, na ordem em que aparecem
- Condição
 - Escolher entre executar ou não executar um conjunto de instruções
- Repetição

Executar, repetidamente, um conjunto de instruções de acordo com uma condição

Estrutura de controlo : Sequência

Problema1: Ler 2 números inteiros indicados pelo utilizador e apresentar a sua soma.

(1) Declarar variáveis, (2) ler dados, (3) somar números e (4) mostrar soma

Pseudocódigo

INICIO

ED: numero1, numero2, soma INTEIRO

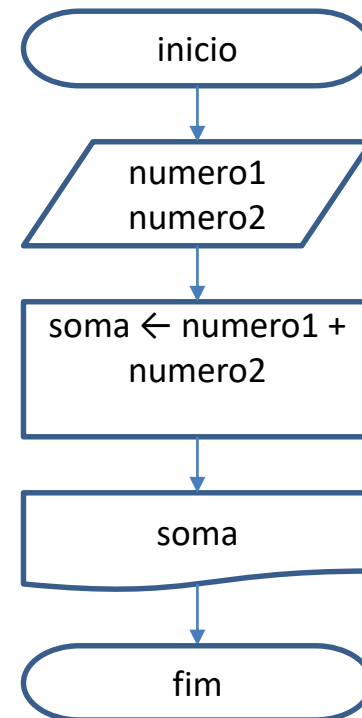
LER (numero1, numero2)

soma \leftarrow numero1 + numero2

ESCREVER (soma)

FIM

Fluxograma



Estrutura de controlo : Sequência

Problema1: Ler 2 números inteiros indicados pelo utilizador e apresentar a sua soma.

INICIO

ED: numero1, numero2, **soma** INTEIRO

LER (numero1, numero2)

soma \leftarrow numero1 + numero2

ESCREVER (**soma**)

FIM

INICIO

ED: numero1, numero2 INTEIRO

LER (numero1)

LER (numero2)

ESCREVER (numero1 + numero2)

FIM

Estrutura de controlo : Sequência

Problema2: Conhecendo a aresta de um cubo calcule o seu volume.

(1) Declarar variáveis, (2) ler aresta, (3) calcular volume e (4) mostrar volume

Pseudocódigo

INICIO

ED: aresta, volume INTEIRO

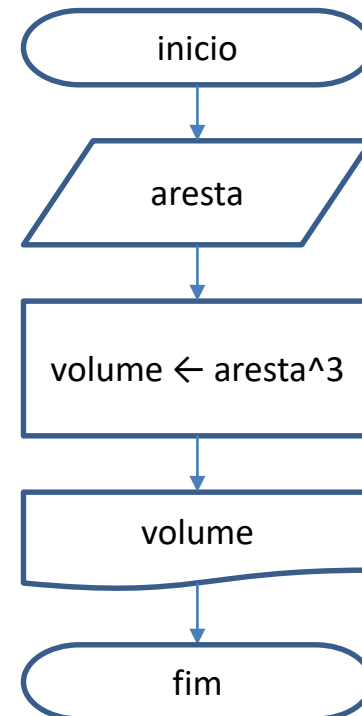
LER (aresta)

$\text{volume} \leftarrow (\text{aresta} ^ 3)$

ESCREVER (volume)

FIM

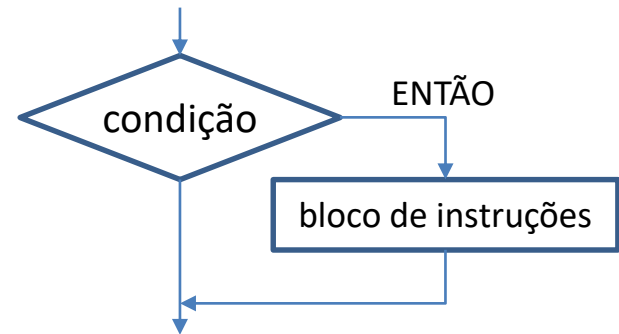
Fluxograma



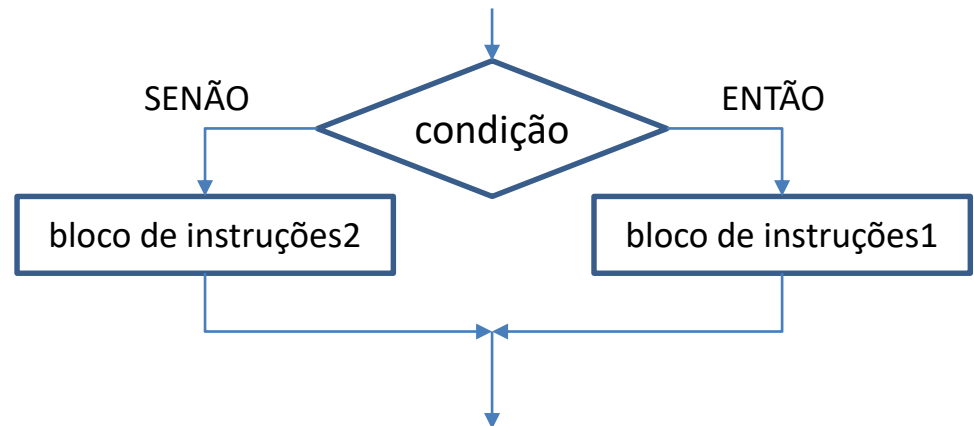
</Sequências>

Estrutura de controlo : Condição

SE (condição) ENTÃO
 <bloco de instruções>
FIMSE



SE (condição) ENTÃO
 <bloco de instruções1>
SENÃO
 <bloco de instruções2>
FIMSE



Estrutura de controlo : Condição

Problema1: Indique se um número lido pelo teclado é zero.

Pseudocódigo

INICIO

ED: numero INTEIRO

LER (numero)

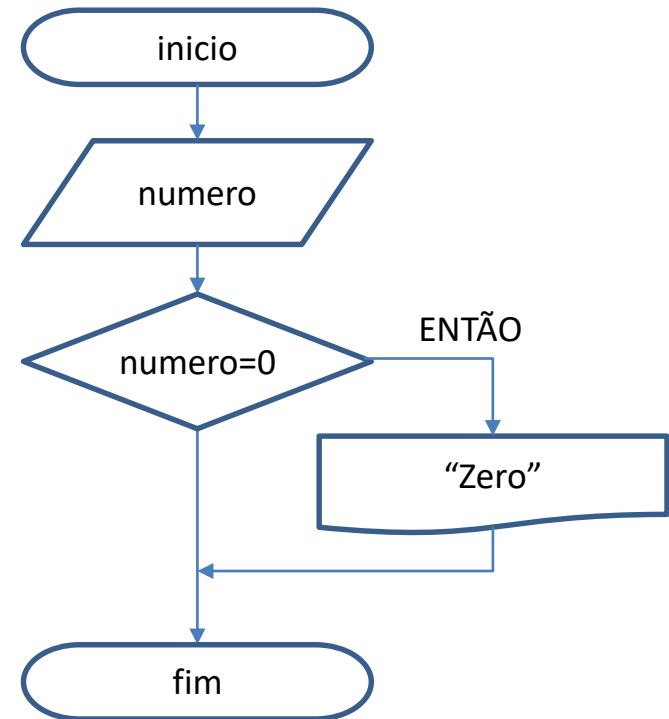
SE (numero = 0) ENTÃO

 ESCREVER ("Zero")

FIMSE

FIM

Fluxograma



Estrutura de controlo : Condição

Problema2: Indique se um número lido pelo teclado é positivo ou negativo.

Pseudocódigo

INICIO

ED: numero INTEIRO

LER (numero)

SE (numero \geq 0) ENTÃO

 ESCREVER ("Positivo")

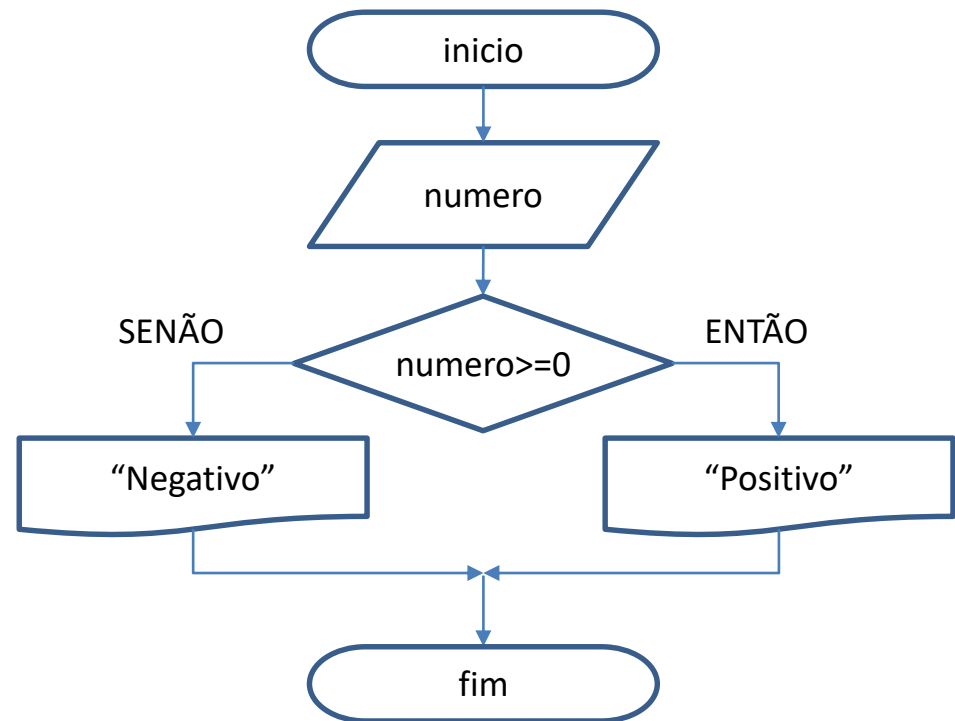
SENÃO

 ESCREVER ("Negativo")

FIMSE

FIM

Fluxograma



Estrutura de controlo : Condição

Condições encaixadas

Problema3: Indique se um número lido pelo teclado é positivo, negativo ou zero.

INICIO

ED: numero INTEIRO

LER (numero)

SE (numero = 0) ENTÃO

 ESCREVER ("Zero")

SENÃO

 SE (numero > 0) ENTÃO

 ESCREVER ("Positivo")

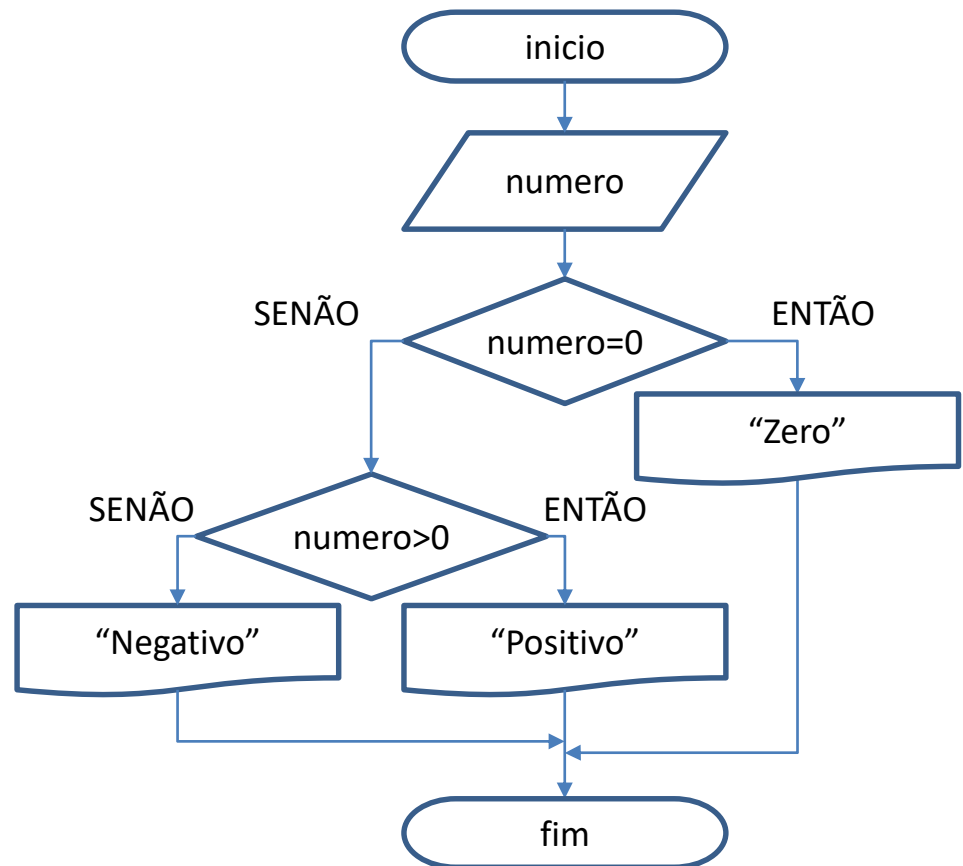
 SENÃO

 ESCREVER ("Negativo")

 FIMSE

FIMSE

FIM



Estrutura de controlo : Condição

Condições encaixadas

Problema4: Escreva um algoritmo capaz de ler dois valores entre 0 e 20 (relativos às duas provas de avaliação que um aluno efetuou) e devolver a situação atual desse aluno com base nos seguintes critérios:

- A nota do aluno é obtida pela média das duas avaliações
- De acordo com a nota, a situação do aluno é:
 - “Reprovado” – se nota inferior a 7.5
 - “Oral” – se nota igual ou superior a 7.5 mas inferior a 10
 - “Aprovado” – se nota igual ou superior a 10

Estrutura de controlo : Condição

Condições encaixadas

Problema4:

INICIO

ED: av1, av2 INTEIRO

ED: nota REAL

LER (av1, av2)

$\text{nota} \leftarrow (\text{av1} + \text{av2}) / 2$

SE (nota < 7.5) ENTÃO

 ESCREVER ("Reprovado")

SENÃO

 SE (nota < 10) ENTÃO

 ESCREVER ("Oral")

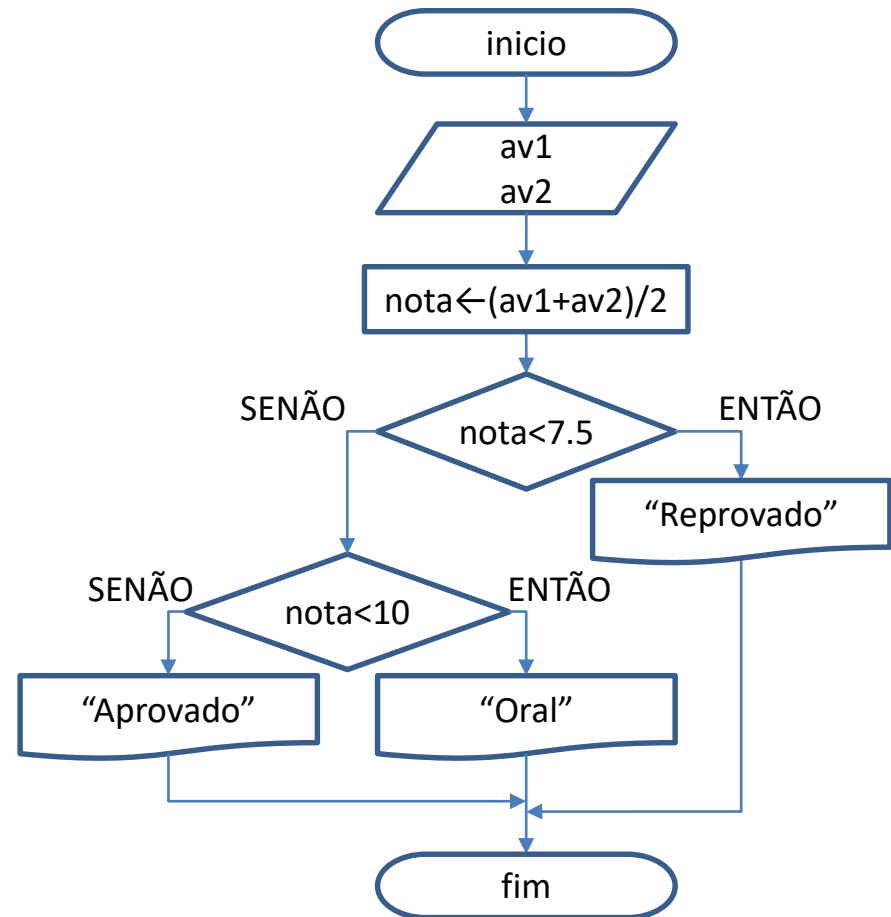
 SENÃO

 ESCREVER ("Aprovado")

 FIMSE

FIMSE

FIM



Estrutura de controlo : Condição

Álgebra Booleana

- Desenvolvida pelo matemático inglês George Boole
- Usada para descrever os circuitos que podem ser construídos pela combinação de portas lógicas, em que as variáveis podem ter apenas valores 0 e 1 (Falso e Verdadeiro)
- Existem apenas três operadores E, OU e NÃO

E	V	F	$X \leftarrow 8$	$(X > 5) \text{ E } (X < 10)$	V
V	V	F		$(X > 5) \text{ E } (X < 7)$	F
F	F	F			
OU	V	F	$X \leftarrow 8$	$(X > 5) \text{ OU } (X < 10)$	V
V	V	V		$(X > 5) \text{ OU } (X < 7)$	V
F	V	F		$(X > 15) \text{ OU } (X < 7)$	F
NAO	V	F	$X \leftarrow 8$	NÃO $(X > 5)$	F
	F	V		NÃO $((X > 15) \text{ OU } (X < 7))$	V

Estrutura de controlo : Condição

Condições encaixadas

Problema4:

INICIO

ED: av1, av2 INTEIRO

ED: nota REAL

LER (av1, av2)

$\text{nota} \leftarrow (\text{av1} + \text{av2}) / 2$

SE (nota < 7.5) ENTÃO

 ESCREVER ("Reprovado")

FIMSE

SE (nota >= 7.5 E nota < 10) ENTÃO

 ESCREVER ("Oral")

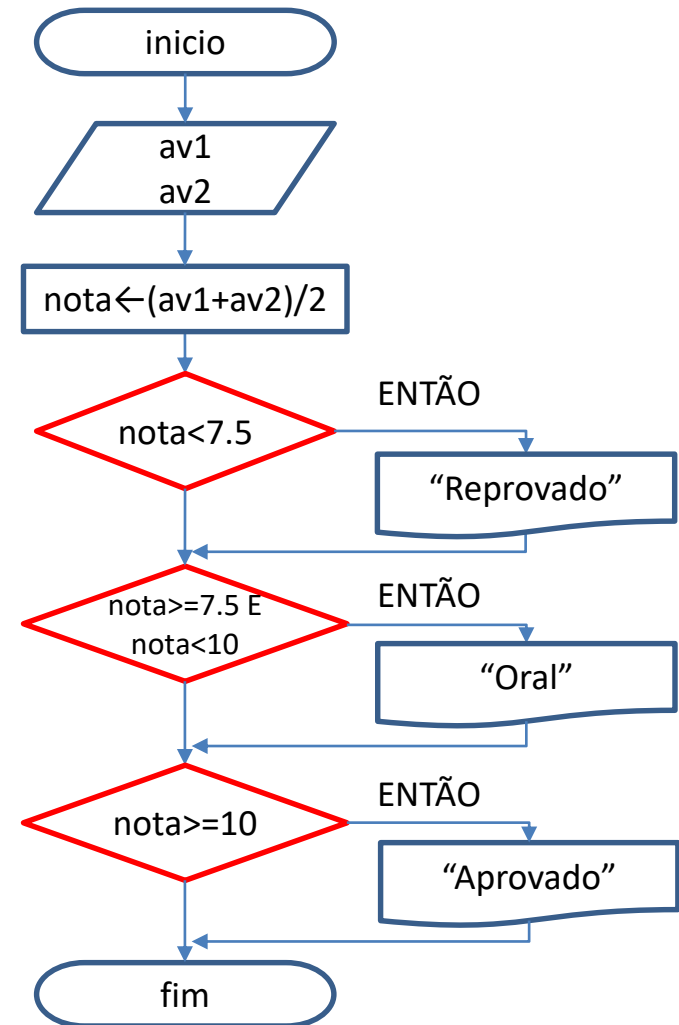
FIMSE

SE (nota >= 10) ENTÃO

 ESCREVER ("Aprovado")

FIMSE

FIM



Conceitos Fundamentais

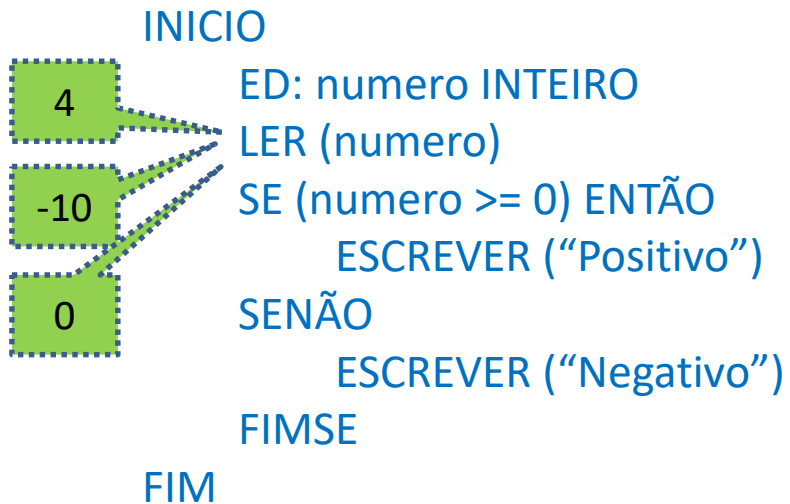
Validação do algoritmo

- Traçagem
 - Verificar se o algoritmo faz o que se pretende
 - Executar manualmente, simulando os passos
 - Registrar os valores das variáveis e acompanhar a evolução das suas modificações

Conceitos Fundamentais

Validação do algoritmo

Problema2: Indique se um número lido pelo teclado é positivo ou negativo.



	<i>numero</i>	<i>numero >= 0</i>
<i>LER (numero)</i>	4	
<i>SE (numero >= 0)</i>		V
<i>ESCREVER ("Positivo")</i>		

	<i>numero</i>	<i>numero >= 0</i>
<i>LER (numero)</i>	-10	
<i>SE (numero >= 0)</i>		F
<i>ESCREVER ("Negativo")</i>		

	<i>numero</i>	<i>numero >= 0</i>
<i>LER (numero)</i>	0	
<i>SE (numero >= 0)</i>		V
<i>ESCREVER ("Positivo")</i>		

</Condições>

Estrutura de controlo : Repetição

REPETIR ENQUANTO (<condição>)

<bloco de instruções>

FIMREPETIR

REPETIR

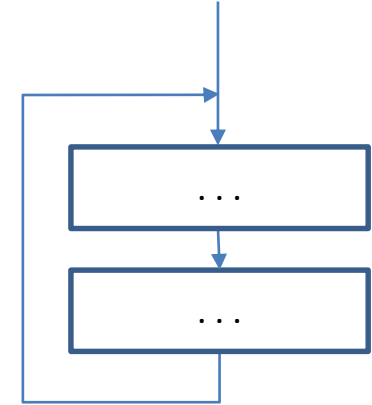
<bloco de instruções>

ENQUANTO (<condição>)

REPETIR PARA <v> \leftarrow <vi> ATE <vf> PASSO <p>

<bloco de instruções>

FIMREPETIR



<v> : variável de controlo
<vi> : valor inicial
<vf> : valor final
<p> : valor do incremento de <v>

Estrutura de controlo : Repetição

Problema1: Escrever os números inteiros de 1 a 100

INICIO

ESCREVER ("1")

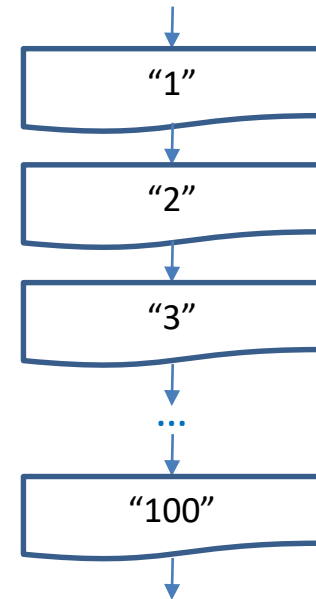
ESCREVER ("2")

ESCREVER ("3")

...

ESCREVER ("100")

FIM



Estrutura de controlo : Repetição

Problema1: Escrever os números inteiros de 1 a 100

INICIO

ED: n INTEIRO

$n \leftarrow 1$

ESCREVER (n)

$n \leftarrow n + 1$

ESCREVER (n)

$n \leftarrow n + 1$

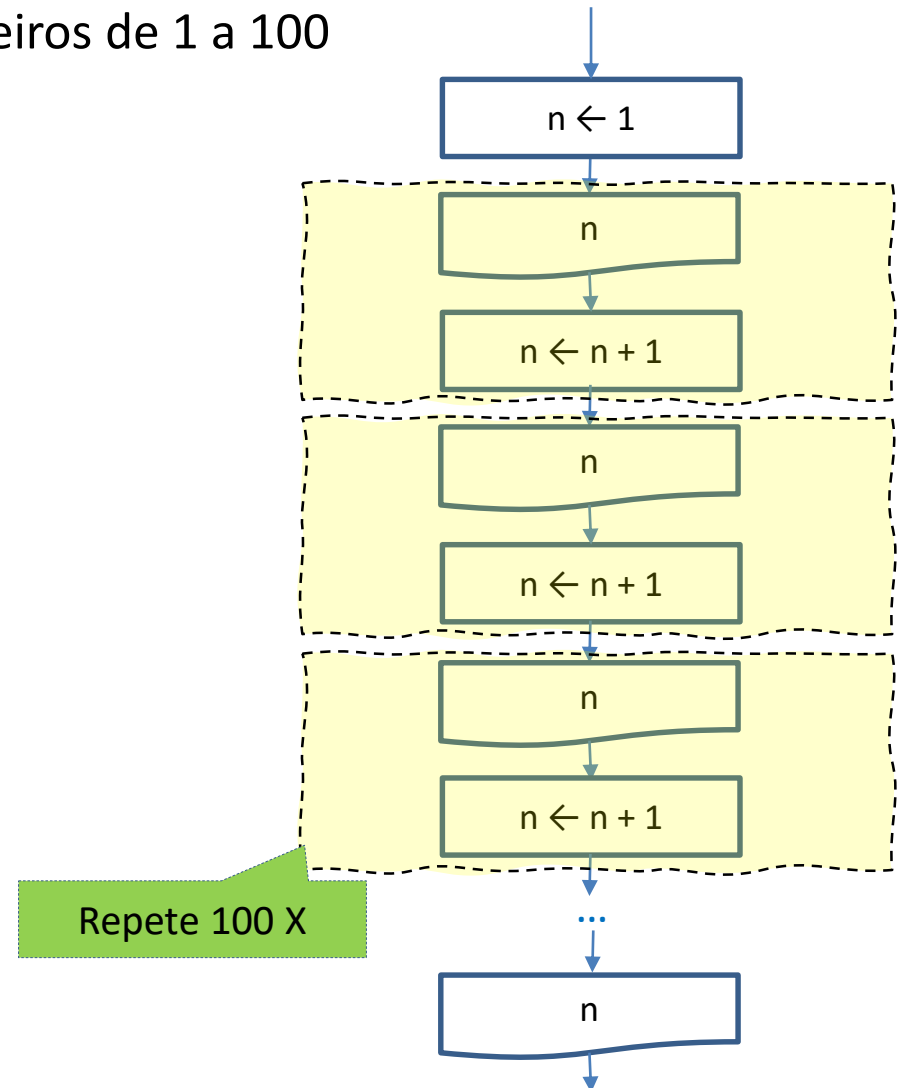
ESCREVER (n)

...

$n \leftarrow n + 1$

ESCREVER (n)

FIM



Estrutura de controlo : Repetição

Problema1: Escrever os números inteiros de 1 a 100

INICIO

ED: n INTEIRO

$n \leftarrow 1$

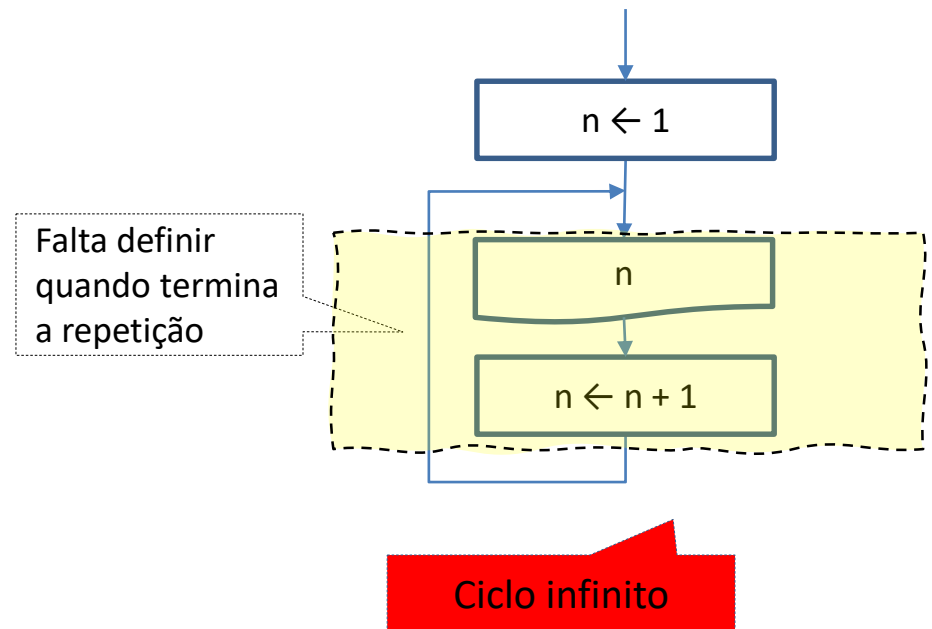
REPETIR

 ESCREVER (n)

$n \leftarrow n + 1$

FIMREPETIR

FIM



Estrutura de controlo : Repetição

REPETIR ENQUANTO ...

REPETIR ENQUANTO (<condição>)

<bloco de instruções>

FIMREPETIR

Problema1: Escrever os números inteiros de 1 a 100

INICIO

ED: n INTEIRO

$n \leftarrow 1$

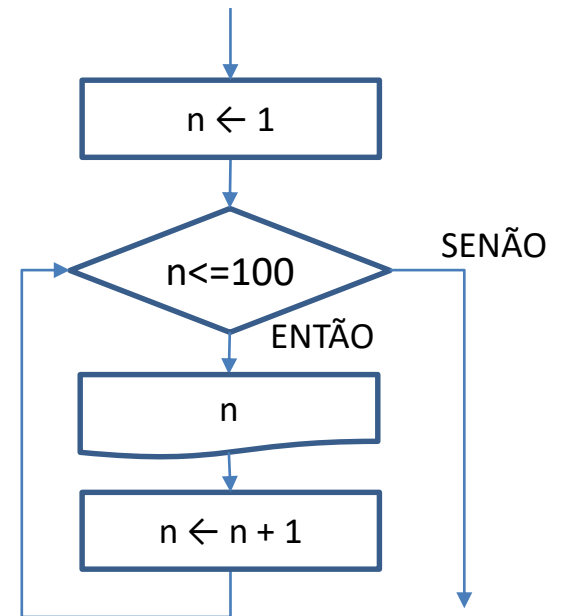
REPETIR ENQUANTO ($n \leq 100$)

 ESCREVER (n)

$n \leftarrow n + 1$

FIMREPETIR

FIM



Estrutura de controlo : Repetição

REPETIR ... ENQUANTO

REPETIR

<bloco de instruções>

ENQUANTO (*<condição>*)

Problema1: Escrever os números inteiros de 1 a 100

INICIO

ED: n INTEIRO

$n \leftarrow 1$

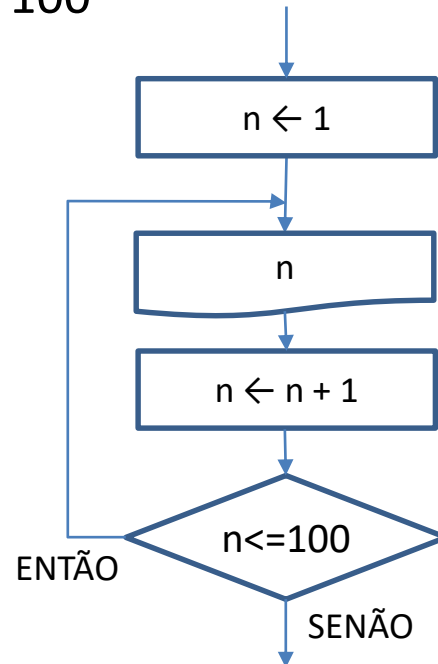
REPETIR

 ESCREVER (n)

$n \leftarrow n + 1$

ENQUANTO ($n \leq 100$)

FIM



Estrutura de controlo : Repetição

REPETIR PARA ...

REPETIR PARA $\langle v \rangle \leftarrow \langle vi \rangle$ ATE $\langle vf \rangle$ PASSO $\langle p \rangle$

$\langle \text{bloco de instruções} \rangle$

FIMREPETIR

$\langle v \rangle$: variável de controlo
 $\langle vi \rangle$: valor inicial
 $\langle vf \rangle$: valor final
 $\langle p \rangle$: valor do incremento de $\langle v \rangle$

Problema1: Escrever os números inteiros de 1 a 100

INICIO

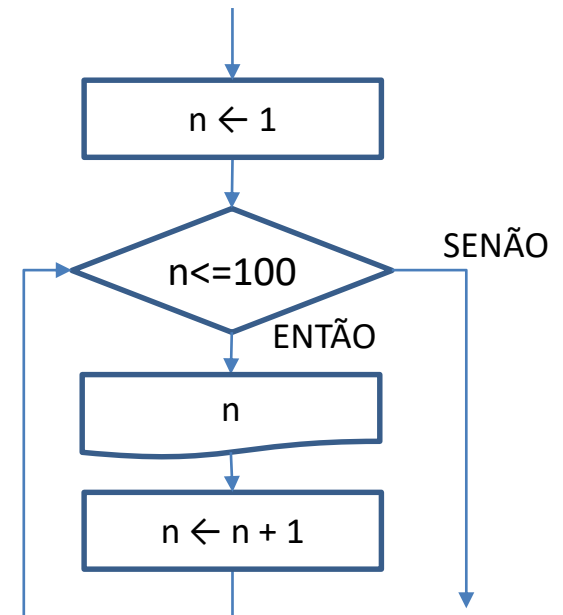
ED: n INTEIRO

REPETIR PARA $n \leftarrow 1$ ATE 100 PASSO 1

ESCREVER (n)

FIMREPETIR

FIM



Estrutura de controlo : Repetição

REPETIR ENQUANTO ...

REPETIR ENQUANTO (<condição>)

<bloco de instruções>

FIMREPETIR

Problema2: Ler e escrever números inteiros enquanto não for inserido o número zero (não imprimir o zero).

INICIO

ED: n INTEIRO

REPETIR ENQUANTO (n != 0)

LER (n)

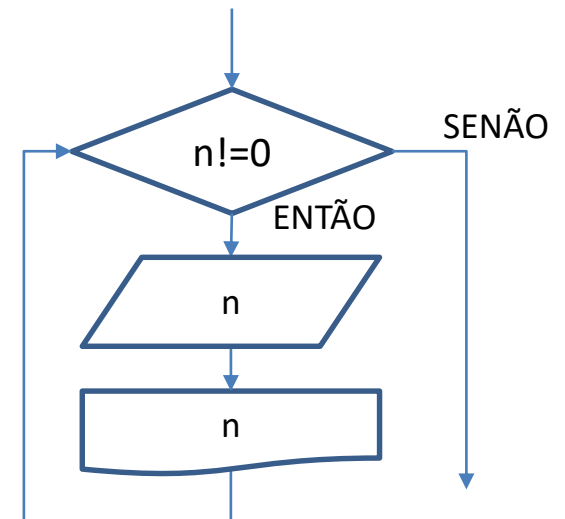
ESCREVER (n)

FIMREPETIR

FIM



Qual o valor de n ?



Estrutura de controlo : Repetição

REPETIR ENQUANTO ...

REPETIR ENQUANTO (<condição>)

<bloco de instruções>

FIMREPETIR

Problema2: Ler e escrever números inteiros enquanto não for inserido o número zero (não imprimir o zero).

INICIO

ED: n INTEIRO

LER (n)

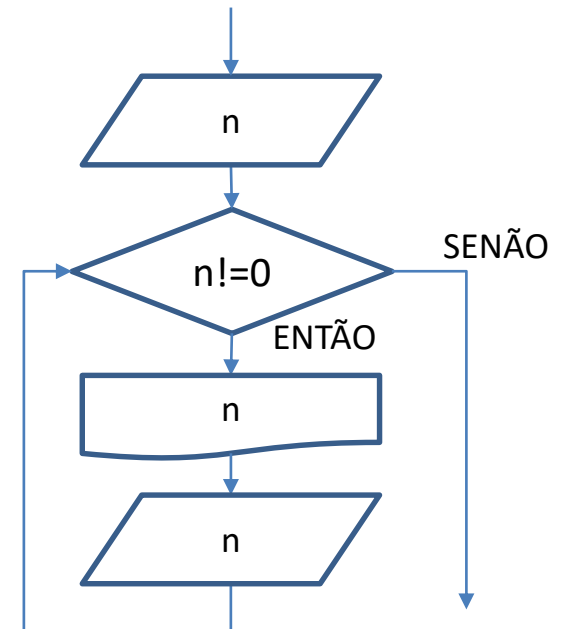
REPETIR ENQUANTO (n != 0)

 ESCREVER (n)

 LER (n)

FIMREPETIR

FIM



Estrutura de controlo : Repetição

REPETIR ... ENQUANTO

REPETIR

<bloco de instruções>

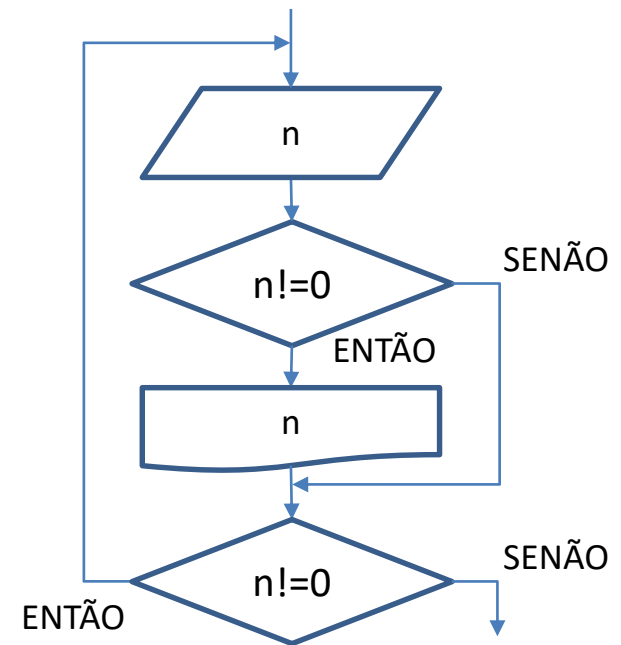
ENQUANTO (*<condição>*)

Problema2: Ler e escrever números inteiros enquanto não for inserido o número zero (não imprimir o zero).

INICIO
ED: n INTEIRO
REPETIR
LER (n)
ESCREVER (n)
ENQUANTO (n != 0)
FIM

Não imprimir o zero

INICIO
ED: n INTEIRO
REPETIR
LER (n)
SE (n != 0) ENTÃO
ESCREVER (n)
FIMSE
ENQUANTO (n != 0)
FIM



Estrutura de controlo : Repetição

REPETIR PARA ...

REPETIR PARA $\langle v \rangle \leftarrow \langle v_i \rangle$ ATE $\langle v_f \rangle$ PASSO $\langle p \rangle$

$\langle \text{bloco de instruções} \rangle$

FIMREPETIR

Problema2: Ler e escrever números inteiros enquanto não for inserido o número zero (não imprimir o zero).

INICIO

ED: n, x INTEIRO

REPETIR PARA x \leftarrow ? ATE ?

...

FIM

- Quantos nº vão ser introduzidos?
- Qual o valor inicial e o final?

Estrutura de controlo : Repetição



Estrutura de controlo : Repetição

Repetições encaixadas

Problema3: Calcular a temperatura média diária ao longo de uma semana. As temperaturas foram registadas hora a hora durante os 7 dias da semana.

INICIO

ED: dia, hora INTEIRO

ED: temperatura, soma, media REAL

REPETIR PARA dia \leftarrow 1 ATE 7 PASSO 1

soma \leftarrow 0

REPETIR PARA hora \leftarrow 1 ATE 24 PASSO 1

LER (temperatura)

soma \leftarrow soma + temperatura

FIMREPETIR

media \leftarrow soma / 24)

ESCREVER (media)

FIMREPETIR

FIM

Para cada dia da
semana

para cada hora
do dia

Estrutura de controlo : Repetição

Problema4: Calcular o fatorial de um número inserido pelo utilizador.

$$n! = 1 * 2 * 3 * \dots * (n-1) * n$$

INICIO

ED: x, n, fatorial INTEIRO

LER (n)

fatorial \leftarrow 1

REPETIR PARA x \leftarrow 1 ATE n PASSO 1

fatorial \leftarrow fatorial * x

FIMREPETIR

ESCREVER (fatorial)

FIM



ou (x \leftarrow 2)

Estrutura de controlo : Repetição

Problema4: Calcular o fatorial de um número inserido pelo utilizador.

$$n! = 1 * 2 * 3 * \dots * (n-1) * n$$

REPETIR ENQUANTO ...

INICIO

ED: x, n, fatorial INTEIRO

LER (n)

fatorial \leftarrow 1

x \leftarrow 1

REPETIR ENQUANTO (x \leq n)

 fatorial \leftarrow fatorial * x

 x \leftarrow x + 1

FIMREPETIR

ESCREVER (fatorial)

FIM

Estrutura de controlo : Repetição

Problema5: Calcular o fatorial de um conjunto de números inseridos pelo utilizador. A leitura termina quando for inserido um número negativo.

$$n! = 1 * 2 * 3 * \dots * (n-1) * n$$

REPETIR PARA ...

INICIO

ED: numero, x, n, fatorial INTEIRO

LER (n)

REPETIR PARA $x \leftarrow ?$ ATE $?$

...

- Quantos n^º vão ser introduzidos?
- Qual o valor inicial e o final?
- Os números vão ser consecutivos?

Estrutura de controlo : Repetição

Problema5: Calcular o fatorial de um conjunto de números inseridos pelo utilizador. A leitura termina quando for inserido um número negativo.

$$n! = 1 * 2 * 3 * \dots * (n-1) * n$$

REPETIR ENQUANTO ...

INICIO

ED: numero, x, n, fatorial INTEIRO

LER (n)

REPETIR ENQUANTO (n >= 0)

fatorial \leftarrow 1

REPETIR PARA x \leftarrow 1 ATE n PASSO 1

fatorial \leftarrow fatorial * x

FIMREPETIR

ESCREVER (fatorial)

LER (n)

FIMREPETIR

FIM

Estrutura de controlo : Repetição

Problema5: Calcular o fatorial de um conjunto de números inseridos pelo utilizador. A leitura termina quando for inserido um número negativo.

$$n! = 1 * 2 * 3 * \dots * (n-1) * n$$

REPETIR ... ENQUANTO

INICIO

ED: numero, x, n, fatorial INTEIRO

REPETIR

LER (n)

SE (n >= 0) ENTAO

fatorial \leftarrow 1

REPETIR PARA x \leftarrow 1 ATE n PASSO 1

fatorial \leftarrow fatorial * x

FIMREPETIR

ESCREVER (fatorial)

FIMSE

ENQUANTO (n >= 0)

FIM

Conceitos Fundamentais

Validação do algoritmo

- Traçagem
 - Verificar se o algoritmo faz o que se pretende
 - Executar manualmente, simulando os passos
 - Registrar os valores das variáveis e acompanhar a evolução das suas modificações

Conceitos Fundamentais

Validação do algoritmo

Problema1: Calcular o fatorial de um número inserido pelo utilizador.

INICIO

ED: $x, n, \text{fatorial}$ INTEIRO

LER (n)

$\text{fatorial} \leftarrow 1$

$x \leftarrow 2$

REPETIR ENQUANTO ($x \leq n$)

$\text{fatorial} \leftarrow \text{fatorial} * x$

$x \leftarrow x + 1$

FIMREPETIR

ESCREVER (fatorial)

FIM

4

	n	fatorial	x	$x \leq n$
LER (n)	4			
$\text{fatorial} \leftarrow 1$		1		
$x \leftarrow 2$			2	
REPETIR ENQUANTO ($x \leq n$)				V
$\text{fatorial} \leftarrow \text{fatorial} * x$		2		
$x \leftarrow x + 1$			3	
REPETIR ENQUANTO ($x \leq n$)				V
$\text{fatorial} \leftarrow \text{fatorial} * x$		6		
$x \leftarrow x + 1$			4	
REPETIR ENQUANTO ($x \leq n$)				V
$\text{fatorial} \leftarrow \text{fatorial} * x$		24		
$x \leftarrow x + 1$			5	
REPETIR ENQUANTO ($x \leq n$)				F
ESCREVER (fatorial)		24		

Estrutura de controlo : Repetição

Problema6: Escrever em sequência os algarismos das unidades, dezenas, ... de um número inteiro.

INICIO

ED: numero INTEIRO
LER(numero)

...

Como separar os algarismos?

0x0056abf1

numero

2719

INTEIRO

Executar divisões inteiras por 10, ou seja, dividir sempre por 10 sem usar casas decimais

2719 | 10
9 271 | 10
1 27 | 10
7 2 | 10
2 0

Pára de dividir quando o
numero chegar a zero

Não há mais algarismos

Estrutura de controlo : Repetição

Problema6: Escrever em sequência os algarismos das unidades, dezenas, ... de um número inteiro.

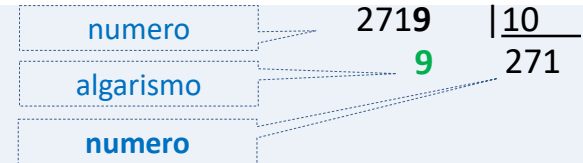
INICIO

ED: numero, algarismo INTEIRO

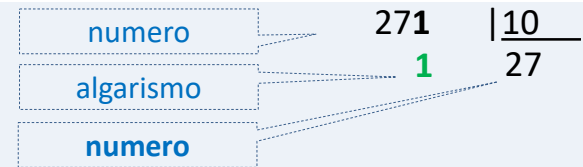
LER(numero)

2719

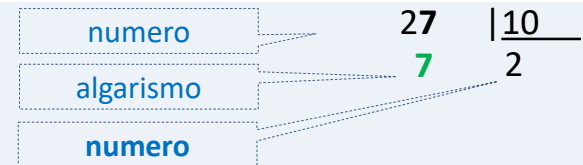
algarismo \leftarrow numero MOD 10
ESCREVER (algarismo)
numero \leftarrow numero DIV 10



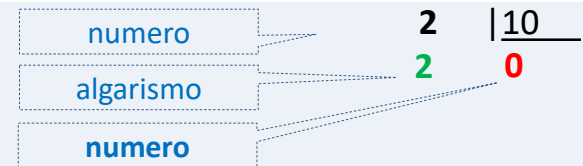
algarismo \leftarrow numero MOD 10
ESCREVER (algarismo)
numero \leftarrow numero DIV 10



algarismo \leftarrow numero MOD 10
ESCREVER (algarismo)
numero \leftarrow numero DIV 10



algarismo \leftarrow numero MOD 10
ESCREVER (algarismo)
numero \leftarrow numero DIV 10



Repete enquanto o numero \neq 0

Estrutura de controlo : Repetição

Problema6: Escrever em sequência os algarismos das unidades, dezenas, ... de um número inteiro. Considere que dispõe das operações DIV e MOD que calculam respectivamente o quociente e o resto da divisão inteira

INICIO

ED: numero, algarismo INTEIRO

LER(numero)

REPETIR

algarismo \leftarrow numero MOD 10

ESCREVER (algarismo)

numero \leftarrow numero DIV 10

ENQUANTO (numero!=0)

FIM

	numero	algarismo	numero!= 0	output
LER (n)	453			
algarismo \leftarrow numero MOD 10		3		
ESCREVER (algarismo)				3
numero \leftarrow numero DIV 10	45			
ENQUANTO (numero!=0)			V	
algarismo \leftarrow n MOD 10		5		
ESCREVER (algarismo)				5
numero \leftarrow numero DIV 10	4			
ENQUANTO (numero!=0)			V	
algarismo \leftarrow numero MOD 10		4		
ESCREVER (algarismo)				4
numero \leftarrow numero DIV 10	0			
ENQUANTO (numero!=0)			F	

Estrutura de controlo : Repetição

Problema7.1: Determinar a quantidade de algarismos de um número inteiro.

INICIO

ED: n, qtd INTEIRO

LER(n)

qtd \leftarrow 0

REPETIR

qtd \leftarrow qtd + 1

n \leftarrow n DIV 10

ENQUANTO (n!=0)

ESCREVER (qtd)

FIM

	n	qtd	n!= 0	output
LER (n)	453			
qtd \leftarrow 0		0		
qtd \leftarrow qtd + 1		1		
n \leftarrow n DIV 10	45			
ENQUANTO (n!=0)			V	
qtd \leftarrow qtd + 1		2		
n \leftarrow n DIV 10	4			
ENQUANTO (n!=0)			V	
qtd \leftarrow qtd + 1		3		
n \leftarrow n DIV 10	0			
ENQUANTO (n!=0)			F	
ESCREVER (qtd)				3

Estrutura de controlo : Repetição

Problema7.2: Determinar a quantidade de algarismos de um número inteiro.

INICIO

ED: n, qtd INTEIRO

LER(n)

qtd \leftarrow 1

REPETIR ENQUANTO (n > 9)

qtd \leftarrow qtd + 1

n \leftarrow n DIV 10

FIM REPETIR

ESCREVER (qtd)

FIM

qualquer número tem pelo menos um algarismo

	n	qtd	n>9	output
LER (n)	453			
qtd \leftarrow 1		1		
REPETIR ENQUANTO (n > 9)			V	
qtd \leftarrow qtd + 1		2		
n \leftarrow n DIV 10	45			
REPETIR ENQUANTO (n > 9)			V	
qtd \leftarrow qtd + 1		3		
n \leftarrow n DIV 10	4			
REPETIR ENQUANTO (n > 9)			F	
ESCREVER (qtd)				3

Estrutura de controlo : Repetição

Problema8.1: Determinar se um número inteiro é composto por mais algarismos pares ou impares.

INICIO

ED: numero, totalAlgarismosPares, totalAlgarismosImpares, algarismo INTEIRO

LER(numero)

Valores iniciais

totalAlgarismosPares \leftarrow 0

totalAlgarismosImpares \leftarrow 0

REPETIR

algarismo \leftarrow numero MOD 10

Capturar algarismo

SE (algarismo MOD 2 = 0) ENTAO

Analisar algarismo

totalAlgarismosPares \leftarrow totalAlgarismosPares + 1

SENAO

totalAlgarismosImpares \leftarrow totalAlgarismosImpares + 1

FIM SE

numero \leftarrow numero DIV 10

Descartar algarismo

ENQUANTO (numero !=0)

SE (totalAlgarismosPares > totalAlgarismosImpares) ENTAO

Analisar resultado

ESCREVER ("existem mais algarismos pares")

SENAO

SE (totalAlgarismosPares = totalAlgarismosImpares) ENTAO

ESCREVER ("existem a mesma quantidade de algarismos pares e impares")

SENAO

ESCREVER ("existem mais algarismos impares")

FIM SE

FIM SE

FIM

Estrutura de controlo : Repetição

Problema8.2: Determinar se um número inteiro é composto por mais algarismos pares ou impares.

INICIO

ED: numero, totalAlgarismosPares, totalAlgarismosImpares, algarismo INTEIRO

LER(numero)

totalAlgarismosPares \leftarrow 0

totalAlgarismosImpares \leftarrow 0

REPETIR ENQUANTO (numero !=0)

algarismo \leftarrow numero MOD 10

SE (algarismo MOD 2) = 0) ENTAO

totalAlgarismosPares \leftarrow totalAlgarismosPares + 1

SENAO

totalAlgarismosImpares \leftarrow totalAlgarismosImpares + 1

FIM SE

numero \leftarrow numero DIV 10

FIM REPETIR

SE (totalAlgarismosPares > totalAlgarismosImpares) ENTAO

ESCREVER (“existem mais algarismos pares”)

SENAO

SE (totalAlgarismosPares = totalAlgarismosImpares) ENTAO

ESCREVER (“existem a mesma quantidade de algarismos pares e impares”)

SENAO

ESCREVER (“existem mais algarismos impares”)

FIM SE

FIM SE

FIM

E se o número inserido for zero ?
Qual o resultado?

Estrutura de controlo : Repetição

Analisar dígitos de um número inteiro

Check digit

“A check digit is a form of redundancy check used for error detection on identification numbers”¹

- UPC (Universal Product Code)
- ISBN (International Standard Book Number)
- EAN (European Article Number)
- IMEI (International Mobile Equipment Identity)
- ...

1 - https://en.wikipedia.org/wiki/Check_digit

Estrutura de controlo : Repetição

“The Luhn algorithm or Luhn formula, also known as the "modulus 10" or "mod 10" algorithm, named after its creator, IBM scientist Hans Peter Luhn, is a simple checksum formula used to validate a variety of identification numbers, such as credit card numbers, IMEI numbers, etc..”¹

1. From the rightmost digit and moving left, double the value of every digit in odd positions. If the result of this doubling operation is greater than 9 (e.g., $8 \times 2 = 16$), then add the digits of the result (e.g., 16: $1 + 6 = 7$, 18: $1 + 8 = 9$) or, alternatively, the same final result can be found by subtracting 9 from that result (e.g., 16: $16 - 9 = 7$, 18: $18 - 9 = 9$).
2. Take the sum of all the digits.
3. The check digit (x) is obtained by computing the sum of the non-check digits then computing 9 times that value modulo 10.

¹ - https://en.wikipedia.org/wiki/Luhn_algorithm

Estrutura de controlo : Repetição

Luhn Algorithm

1. From the rightmost digit and moving left, double the value of every digit in odd positions. If the result of this doubling operation is greater than 9 (e.g., $8 \times 2 = 16$), then add the digits of the result (e.g., 16: $1 + 6 = 7$, 18: $1 + 8 = 9$) or, alternatively, the same final result can be found by subtracting 9 from that result (e.g., 16: $16 - 9 = 7$, 18: $18 - 9 = 9$).
2. Take the sum of all the digits.
3. The check digit (x) is obtained by computing the sum of the non-check digits then computing 9 times that value modulo 10.

Assume an example of an account number "7992739871" that will have a check digit added, making it of the form 7992739871x:

Account number	7	9	9	2	7	3	9	8	7	1	x
Double every other	7	18	9	4	7	6	9	16	7	2	x
Sum digits	7	9	9	4	7	6	9	7	7	2	x

Estrutura de controlo : Repetição

Luhn Algorithm

1. The sum of all the digits in the third row is $67 + x$.
2. Compute the sum of the non-check digits (67).
3. Multiply by 9 ($67 \times 9 = 603$).
4. The check digit (x) is obtained by computing the sum of the non-check digits then computing 9 times that value modulo 10 ($603 \bmod 10 = 3$).
5. The units digit (3) is the check digit. Thus, $x=3$.

Account number	7	9	9	2	7	3	9	8	7	1	x
Double every other	7	18	9	4	7	6	9	16	7	2	x
Sum digits	7	9	9	4	7	6	9	7	7	2	x

Estrutura de controlo : Repetição

Problema9: Calcular o check digit de um número pelo algoritmo de Luhn.

INICIO

ED: numero, posicao, soma, digito, dobro INTEIRO

LER(numero)

posicao \leftarrow 0

soma \leftarrow 0

REPETIR ENQUANTO (numero != 0)

pára quando o numero chegar a zero

digito \leftarrow numero MOD 10

numero \leftarrow numero DIV 10

retira o algarismo à direita no número

posicao \leftarrow posicao + 1

SE (posicao MOD 2 != 0) ENTAO

dobro \leftarrow digito * 2

SE(dobro > 9) ENTAO

dobro \leftarrow dobro - 9

FIMSE

processa com base no algarismo

soma \leftarrow soma + dobro

SENAO

soma \leftarrow soma + digito

FIMSE

FIMREPETIR

ESCREVER("check digit=", (soma * 9) MOD 10)

FIM

</Repetições>