



Execução de instruções

Aula nº4

Sumário

- Funções
- Seleccção simples de instruções
- Operadores lógicos
- Seleccção composta de instruções
- Seleccção múltipla
- Exercícios

Algoritmos computacionais

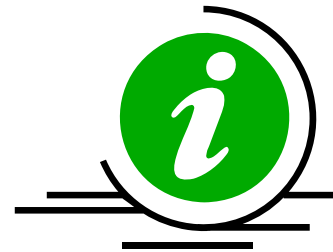
- Acções de um algoritmo computacional

- ▢ Manipular dados

- ▢ Receber dados

- ▢ Guardar dados

- ▢ Imprimir dados



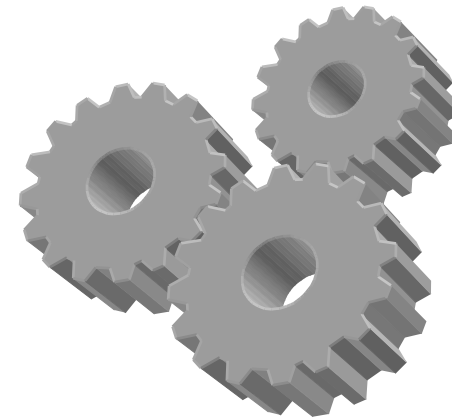
- ▢ Executar instruções

- ▢ Fazer operações aritméticas

- ▢ Fazer operações lógicas

- ▢ Escolha entre várias instruções.

- ▢ Repetir um conjunto de instruções



Operadores Aritméticos no Portugol

Inteiros

- + Adição
- Subtracção
- / Divisão inteira
- * Multiplicação
- % Resto da divisão inteira
- ^ Potenciação

Reais

- + Adição
- Subtracção
- / Divisão inteira
- * Multiplicação
- ^ Potenciação

Texto

- + Concatenação de texto

Fluxo de dados

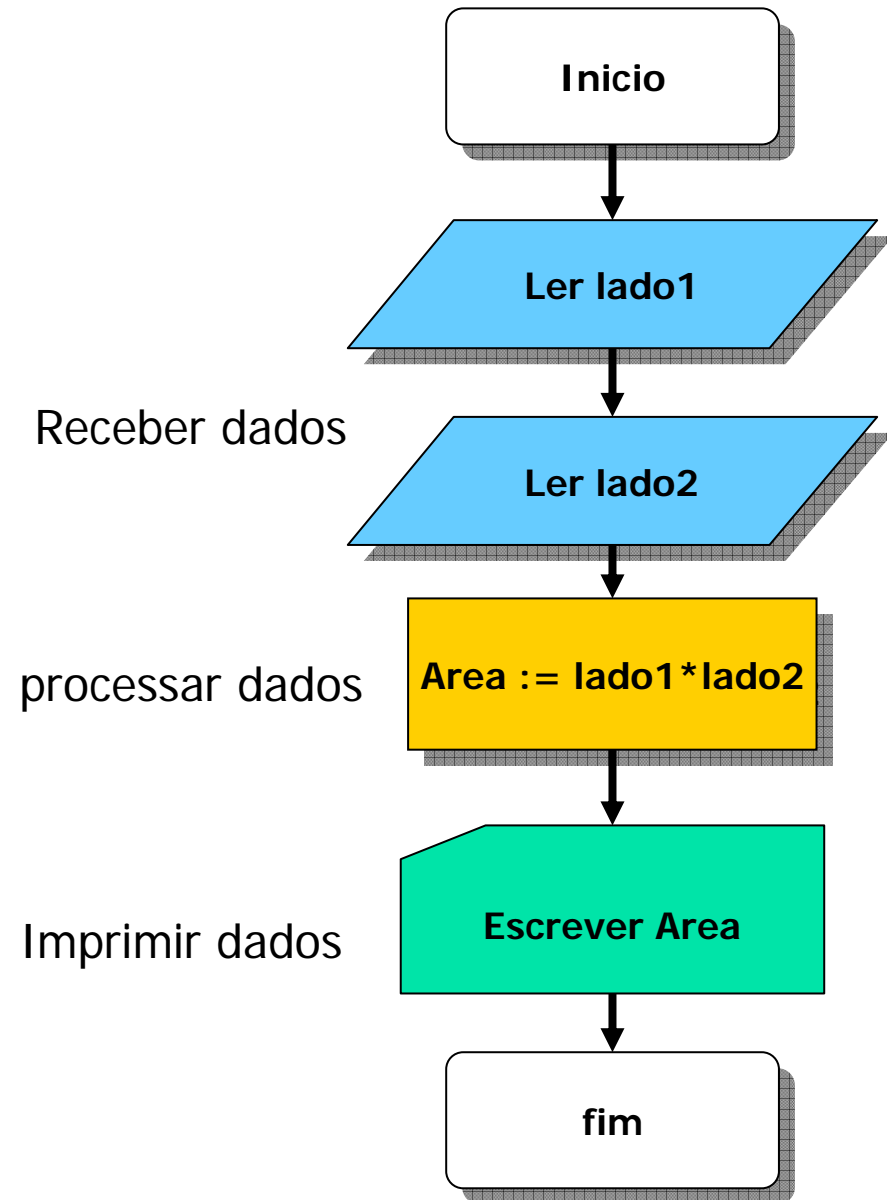
Programa Área do rectângulo

```

inicio
real lado1, lado2, area
ler lado1
ler lado2
area <- lado1 * lado2
escrever area
fim

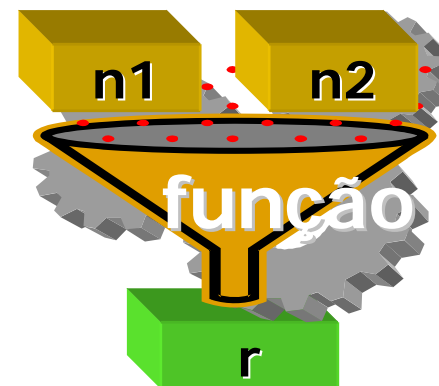
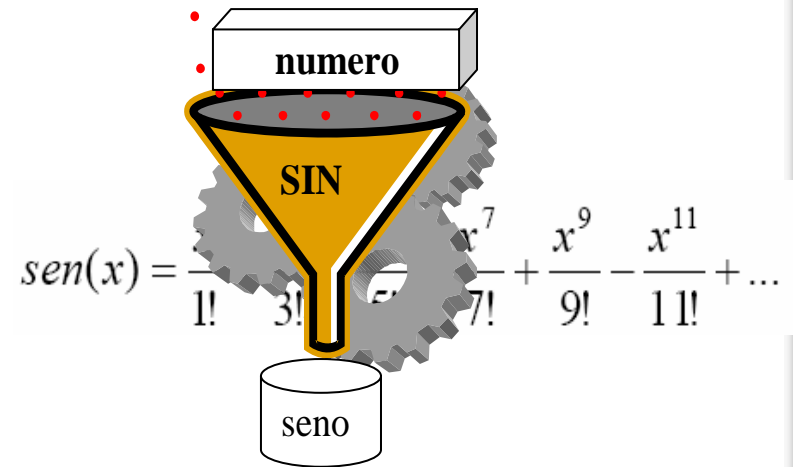
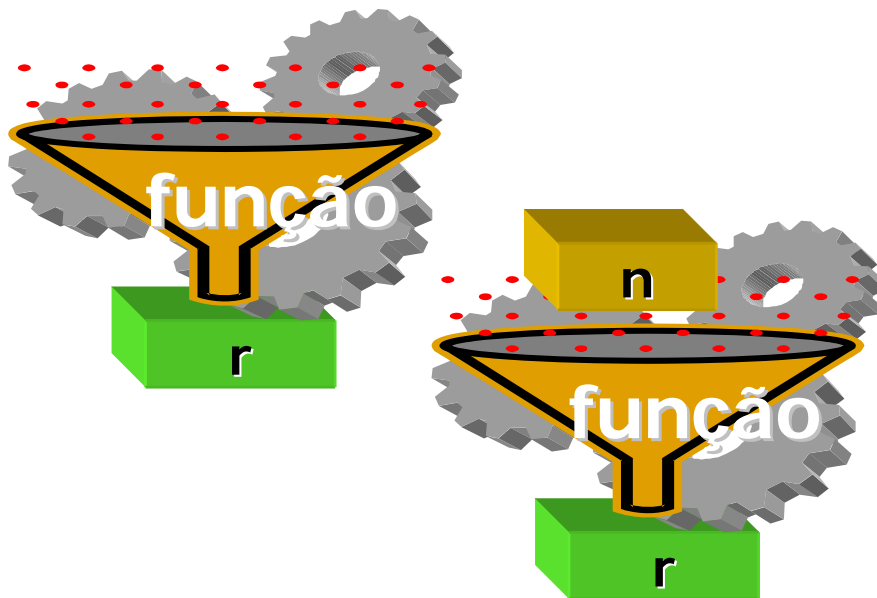
```

area	200
lado2	20
lado1	10



Funções

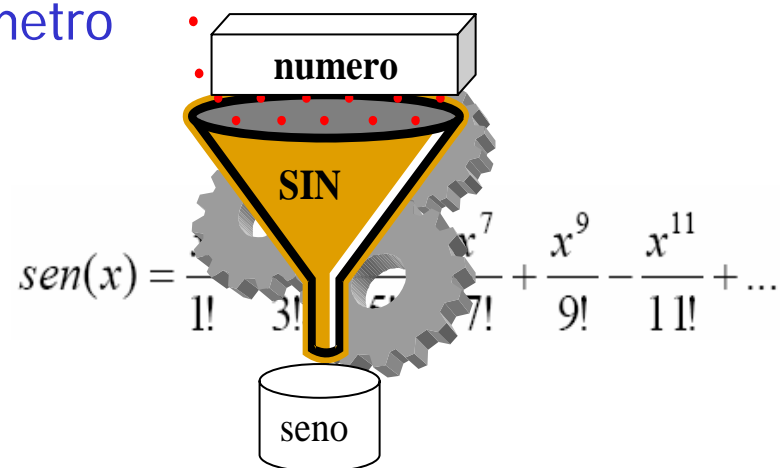
- Fazem cálculos complexos
- Podem ter parâmetros
 - Passados dentro de parêntesis
- Retornam um resultado



Funções do Portugol

■ Funções Trigonométricas com 1 parâmetro

- **SEN** Seno
- **COS** Coseno
- **TAN** Tangente
- **CTG** Cotangente
- **ASEN** Arco Seno
- **ACOS** Arco Coseno
- **ATAN** Arco Tangente
- **ACTG** Arco Cotangent
- **SENH** Seno hiperbolico
- **COSH** Coseno hiperbólico
- **TANH** Tangente hiperbólica
- **CTGH** Cotangente hiperbólica



■ Funções sem parâmetros

- **aleatorio**
 - devolve um número aleatório entre 0 e 1

■ Funções Matemáticas com 1 parâmetro

- **EXP** exponenciação
- **ABS** valor absoluto
- **RAIZ** raiz quadrada
- **LOG** Logaritmo base 10
- **LN** logaritmo base E
- **INT** parte inteira de um número real
- **FRAC** parte fraccionária de um número real
- **ARRED** Arredondamento um número real

Exercício

- Construa um programa que solicite ao utilizador o valor de um ângulo em radianos e de seguida imprima o seu seno, o seu coseno e a sua tangente

Programa test_funcao

```
inicio
  real angulo
  real vseno , vcoseno , vtangente
  escrever "angulo\t:"
  ler angulo
  vseno <- sen ( angulo )
  vcoseno <- cos ( angulo )
  vtangente <- tan ( angulo )
  escrever "seno \t\t:", vseno , "\n"
  escrever "coseno \t\t:", vcoseno , "\n"
  escrever "tangente \t\t:", vtangente , "\n"
fim
```



Exercício

- Construa um programa que solicite ao utilizador o valor de um ângulo **em graus** e de seguida imprima o seu seno, o seu coseno e a sua tangente



Exercício

- Construa um programa que escreve no monitor seis números aleatório para jogar no totoloto

Programa Totoloto

inicio

```
escrever aleatorio () * 49 + 1 , "\n"
```

```
escrever aleatorio () * 49 + 1 , "\n"
```

```
escrever aleatorio () * 49 + 1 , "\n"
```

```
escrever aleatorio () * 49 + 1 , "\n"
```

```
escrever aleatorio () * 49 + 1 , "\n"
```

```
escrever aleatorio () * 49 + 1
```

fim

17.99180703133173

2.589963185286182

31.225030308773675

34.96870617589329

14.44072266602874

46.26479700892334

aleatorio - devolve um número aleatório entre 0 e 1

Exercício

- Construa um programa que escreve no monitor seis números aleatório para jogar no totoloto

Programa Totoloto

```
inicio
escrever int ( aleatorio () * 49 + 1 ) , "\t"
escrever int ( aleatorio () * 49 + 1 ) , "\t"
escrever int ( aleatorio () * 49 + 1 ) , "\t"
escrever int ( aleatorio () * 49 + 1 ) , "\t"
escrever int ( aleatorio () * 49 + 1 ) , "\t"
escrever int ( aleatorio () * 49 + 1 ) , "\t"
fim
```

10 25 7 40 41 9 35

INT parte inteira de um número real



Estruturas de Decisão

Exercício

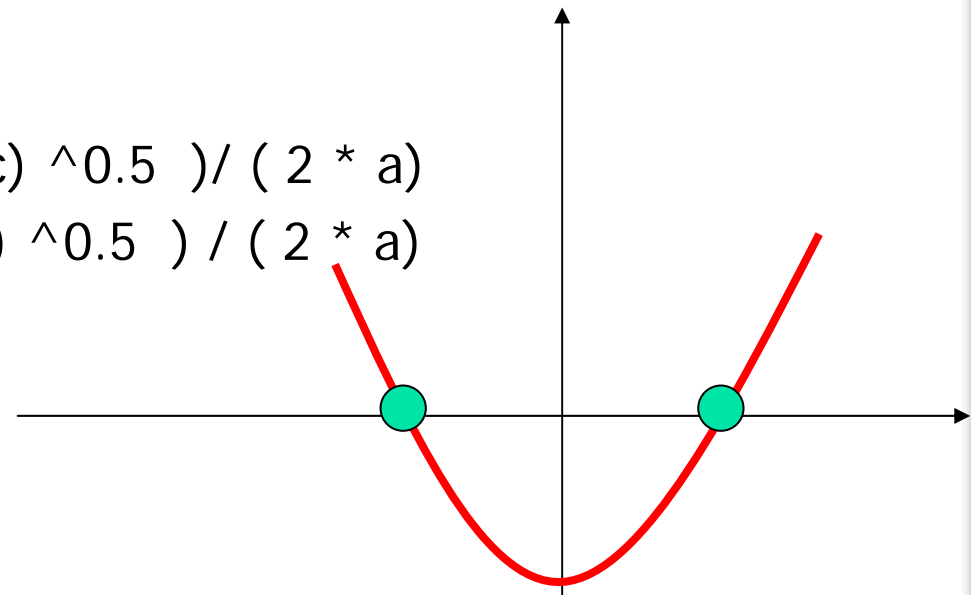
- Construir um programa que calcule os zeros de uma equação de 2º grau cujos coeficientes são introduzidos pelo utilizador

$$ax^2 + bx + c = 0$$

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

- Computacional

- $X1 := (-b + (b^2 - 4*a*c)^{0.5}) / (2 * a)$
- $X2 := (-b - (b^2 - 4*a*c)^{0.5}) / (2 * a)$



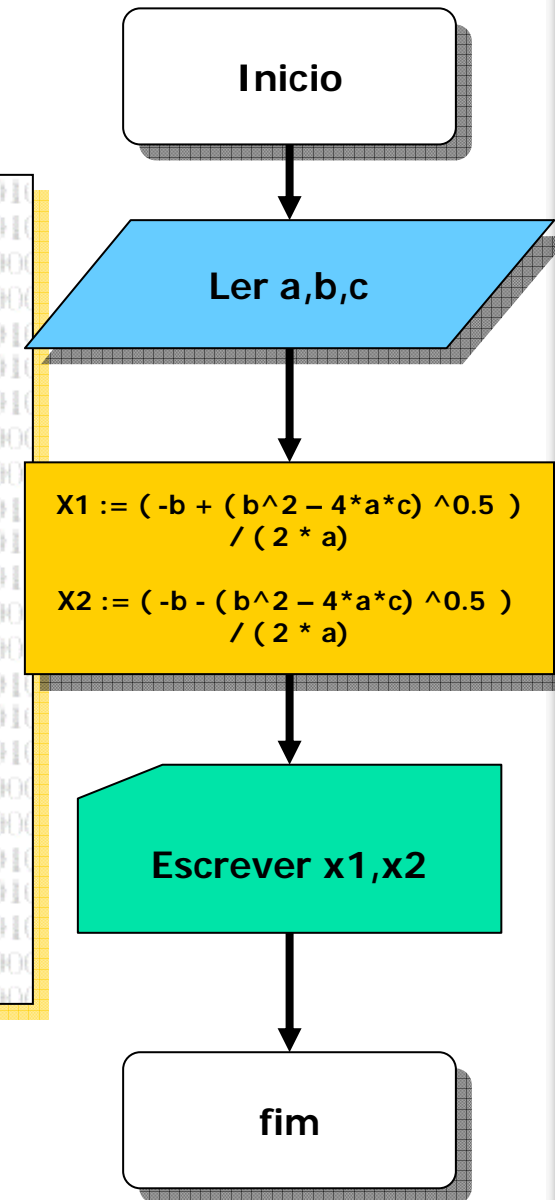
Exercício

Programa Equacao2g

```

inicio
  real a , b , c
  real x1 , x2
  escrever "Raizes da equação de 2º grau\n"
  escrever " a = "
  ler a
  escrever " b = "
  ler b
  escrever " c = "
  ler c
  x1 <- (-b - (b ^ 2 - 4 * a * c) ^ 0.5) / (2 * a)
  x2 <- (-b + (b ^ 2 - 4 * a * c) ^ 0.5) / (2 * a)
  escrever "\n x1 = " , x1 , " x2 = " , x2
fim

```

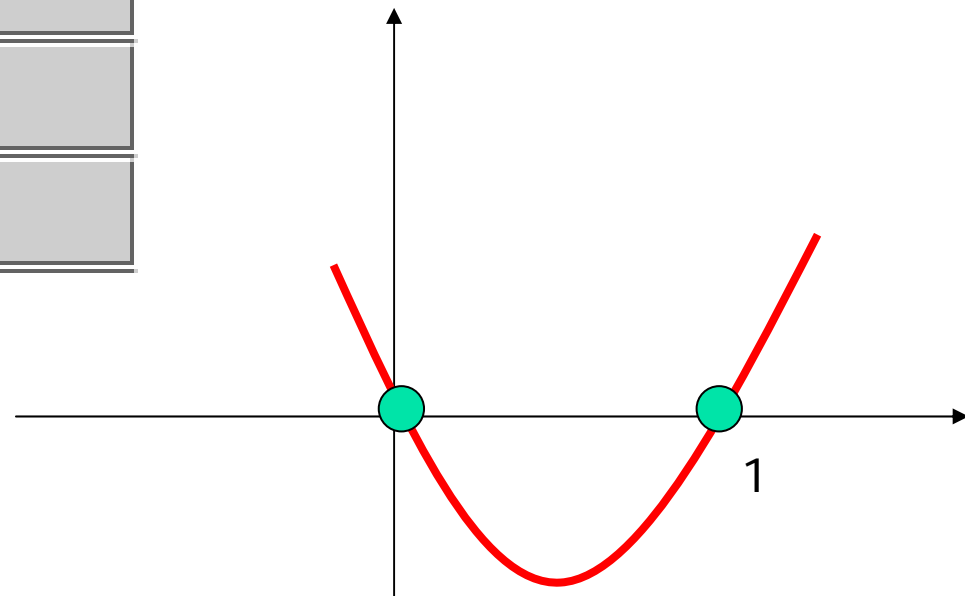


Verificação do algoritmo

x2	1
x1	0
c	0
b	-1
a	1

$$x^2 - x = 0$$

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

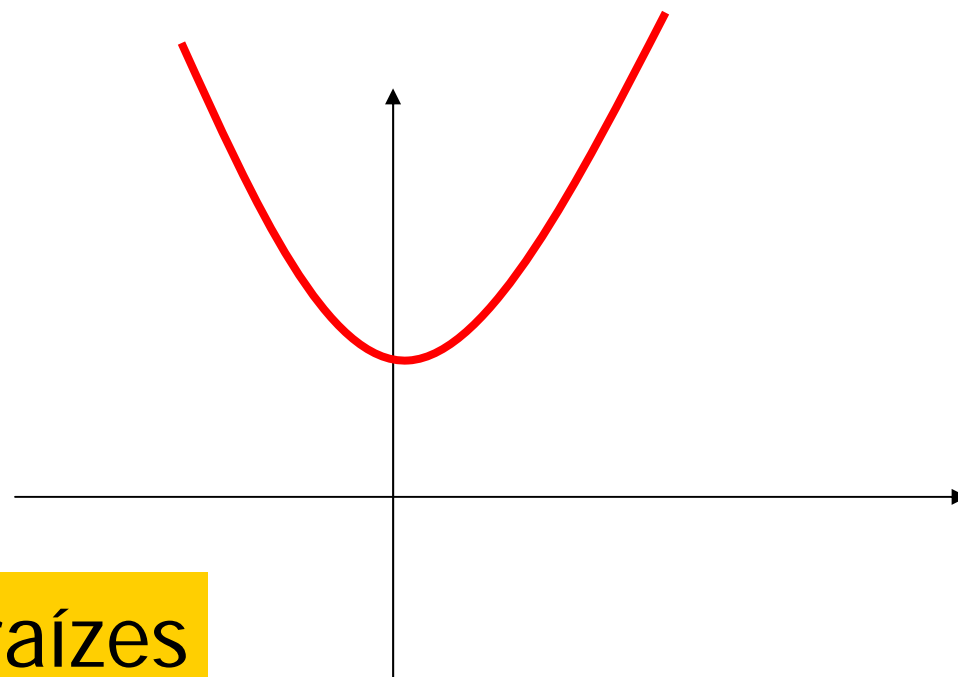


Verificação do algoritmo

x2	□
x1	□
c	3
b	0
a	1

$$x^2 + 3 = 0$$

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$



Não tem raízes

Fluxo de execução

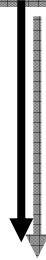
Início / Fim

Entrada de dados

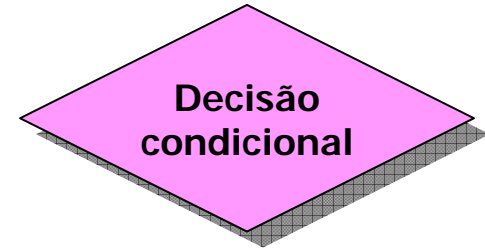
Processamento
(cálculos)

Impressão de dados

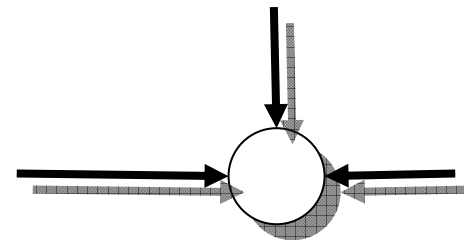
Linha de fluxo



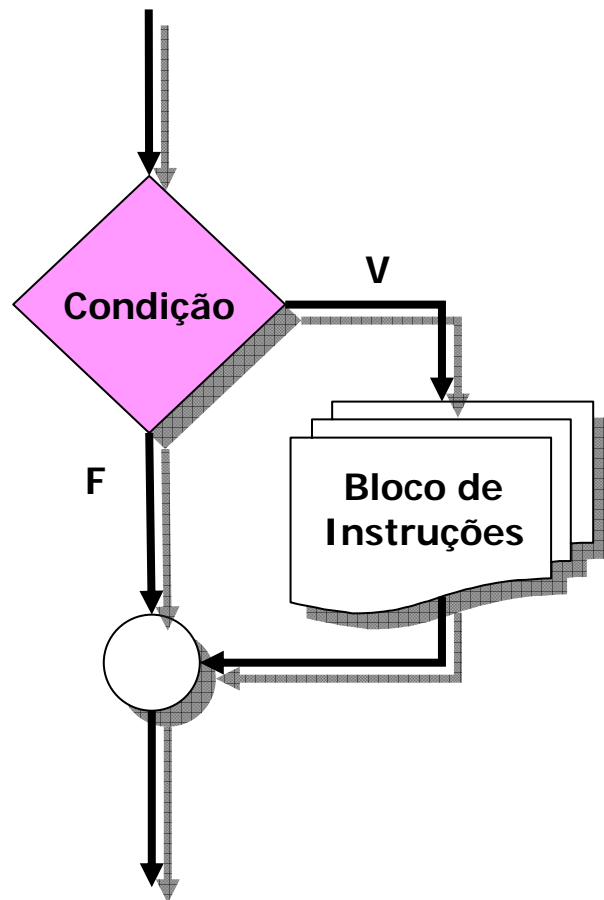
Decisão
condicional



Conector de fluxo



Seleccção Simples



Linguagem estruturada

```
Se condicao Entao  
  intrução 1  
  intrução 2  
  intrução 3  
  ...  
FimSe
```

O resultado das condições é sempre:

- Verdadeiro

OU

- Falso

PRINCÍPIO DO 3º EXCLUÍDO

Operadores lógicos

■ RELACIONAIS

- Maior
 - >
- Menor
 - <
- Igual
 - =
- Maior ou igual
 - >=
- Menor ou igual
 - <=
- Diferente
 - !=

O resultado dos operadores lógicos tem apenas dois valores

- Verdadeiro
- falso

Exemplos

5 < 4

5 > 4

5 != 4

delta < 0

0 < delta

comprimento = altura

Operadores relacionais

Programa Operador_Relacional

```
inicio
inteiro x1, x2
logico iguais
escrever " x1: "
ler x1
escrever " x2 : "
ler x2
iguais <- x1 = x2
escrever "\nresultado x1 = x2 \t", iguais
escrever "\nresultado x1 > x2 \t", x1 > x2
escrever "\nresultado x1 >= x2 \t", x1 >= x2
escrever "\nresultado x1 < x2 \t", x1 < x2
escrever "\nresultado x1 <= x2 \t", x1 <= x2
escrever "\nresultado x1 /= x2 \t", x1 /= x2
fim
```

Operadores lógicos

■ LÓGICOS

- **Conjunção**
 - E
- **Disjunção**
 - OU
- **Negação**
 - NAO

E	0	1
0	0	0
1	0	1

Multiplicação
Lógica

OU	0	1
0	0	1
1	1	1

Soma
Lógica

NAO	0	1
0	1	0
1	0	1

Negação
Lógica

Prioridade dos operadores

Relacionais -> Negação -> Conjunção -> Disjunção

Maior

Menor

Operadores relacionais

Programa Operador_logico

```
inicio
logico x1 , x2
escrever " x1:"
ler x1
escrever " x2: "
ler x2
escrever "\n x1 E x2 \t" , x1 e x2
escrever "\n x1 OU x2 \t" , x1 ou x2
escrever "\n NAO x1 \t" , nao x1
escrever "\n NAO x2 \t" , nao x2
fim
```

Exercícios

E	0	1
0	0	0
1	0	1

Multiplicação
Lógica

OU	0	1
0	0	1
1	1	1

Soma
Lógica

NAO	0	1
	1	0

Negação
Lógica

Exemplos

$2 < 3$ E $3 < 5$

$2 < 3$ E $4 < 2$

$2 < 3$ OU $3 < 4$ E $4 > 5$

NAO($2 =/= 3$) E NAO($4 > 3$)

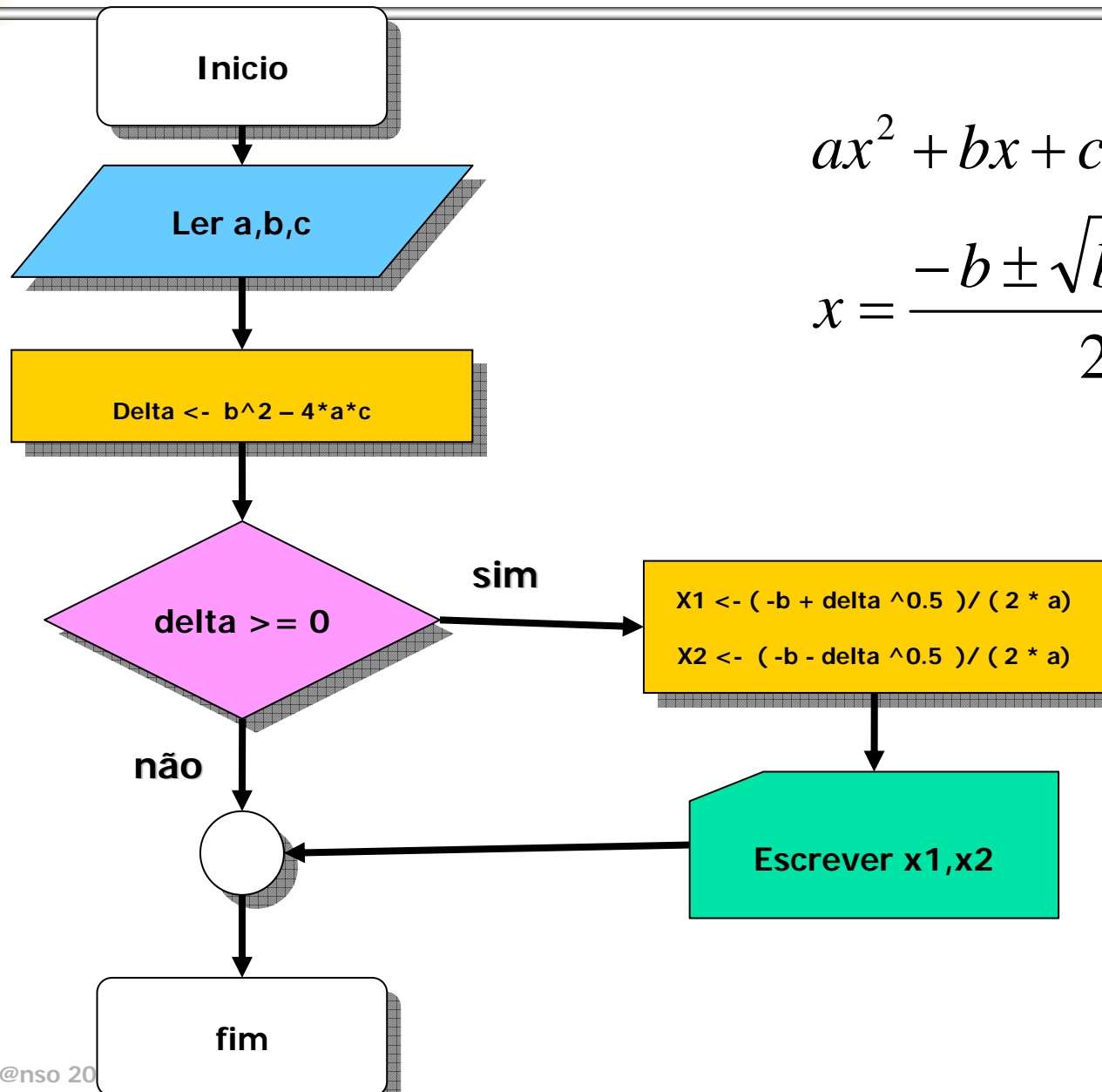
Verdadeiro

Falso

Verdadeiro

Falso

Fluxograma - eq2g



$$ax^2 + bx + c = 0$$

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Exercício

Programa Equacao2g

inicio

real a , b , c , x1 , x2 , delta

escrever "Raizes da equação de 2º grau"

escrever "\na = "

ler a

escrever "b = "

ler b

escrever "c = "

ler c

delta <- b ^ 2 - 4 * a * c

se delta >= 0 entao

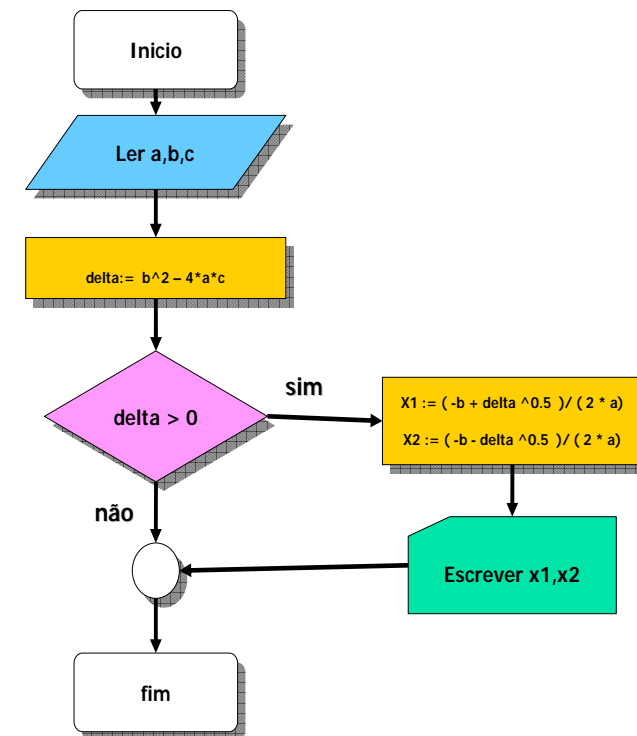
x1 <- (-b - delta ^ 0.5) / (2 * a)

x2 <- (-b + delta ^ 0.5) / (2 * a)

escrever " x1 = ", x1 , " x2 = ", x2

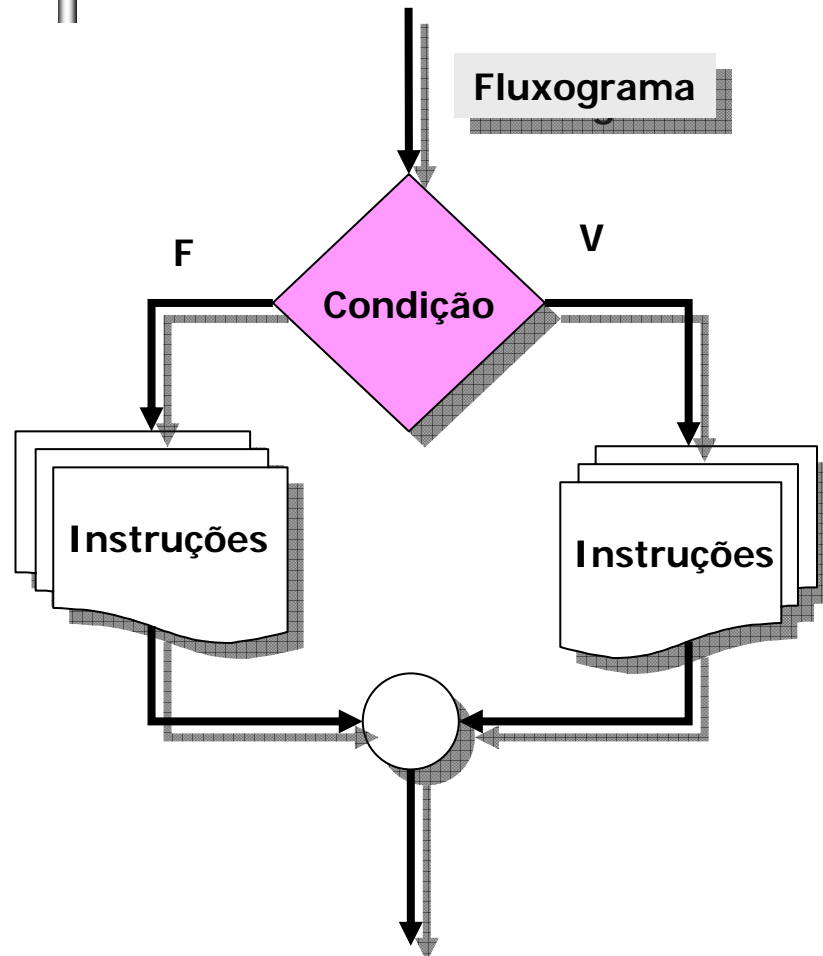
fimse

fim



Identação

Seleção em Alternativa



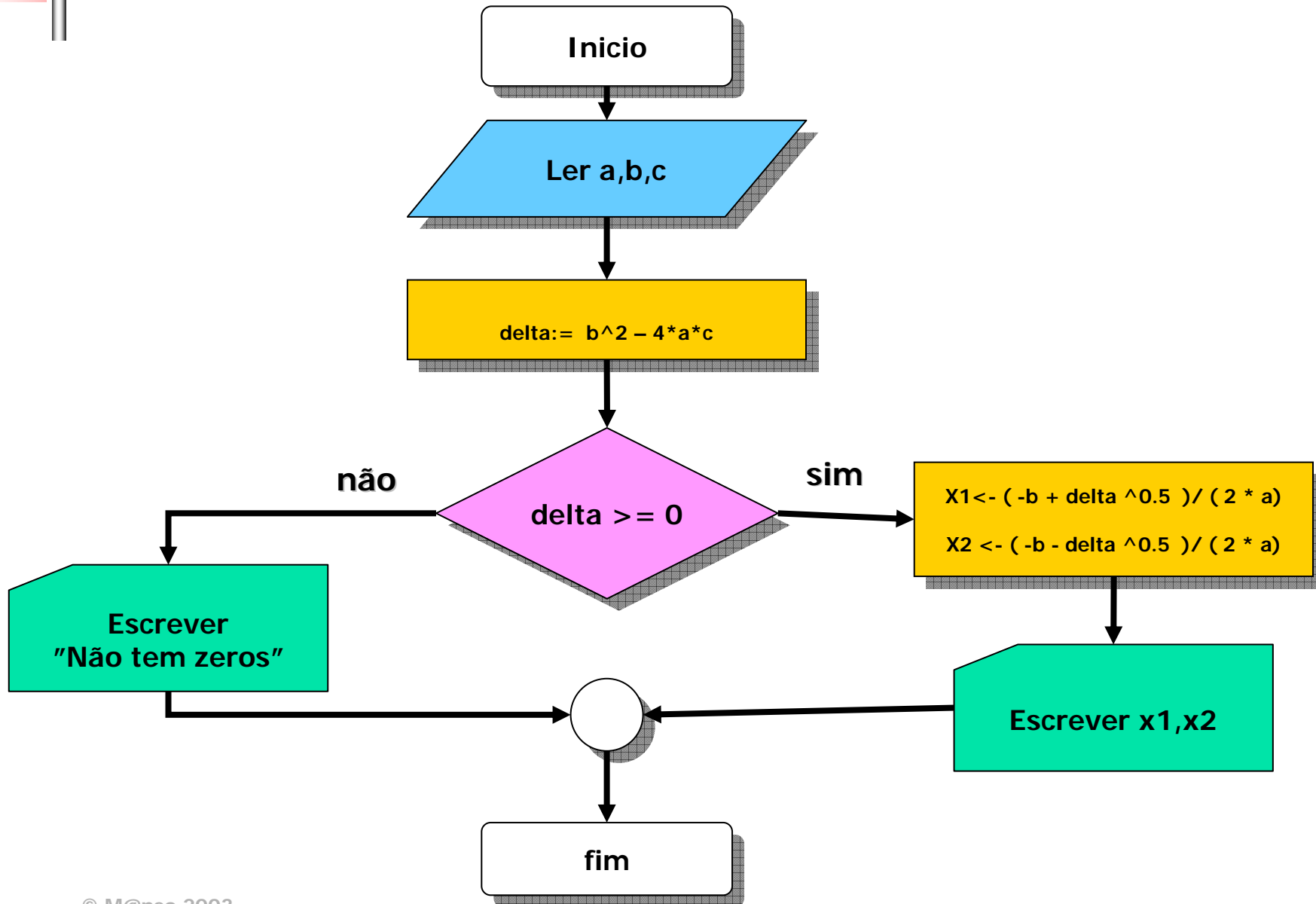
Linguagem estruturada

**Se condicao Entao
instrução**

**Senao
instrução**

Fimse

Fluxograma - eq2g

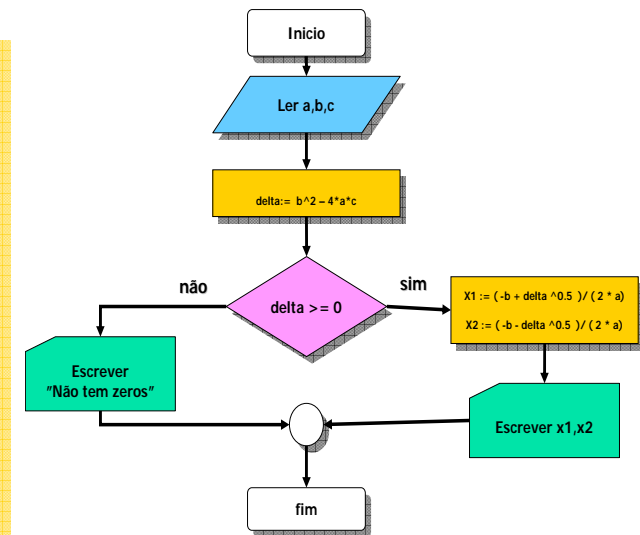


Exercício

Programa Equacao2g

```

variaveis
  numerico a,b,c, x1,x2, delta
inicio
  escrever "Raizes da equação de 2º grau"
  escrever " a = "
  ler a
  escrever " b = "
  ler b
  escrever " c = "
  ler c
  delta := b^2 - 4*a*c
  se delta >= 0 entao
    x1 <- (-b - delta^0.5) / (2*a)
    x2 <- (-b + delta^0.5) / (2*a)
    escrever " x1 = ", x1 , " x2 = ", x2
  senao
    escrever "Não tem zeros"
  fim_se
fim
  
```



Identação

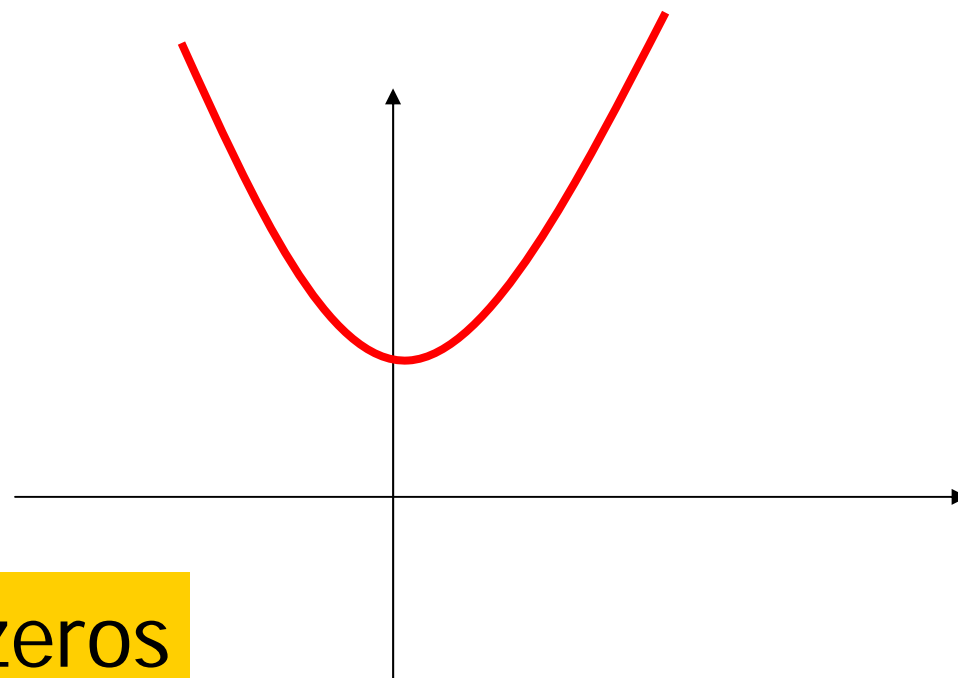
Identação

Verificação do algoritmo

delta	-12
x2	0
x1	0
c	3
b	0
a	1

$$x^2 + 3 = 0$$

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$



Não tem zeros

Verificação do algoritmo

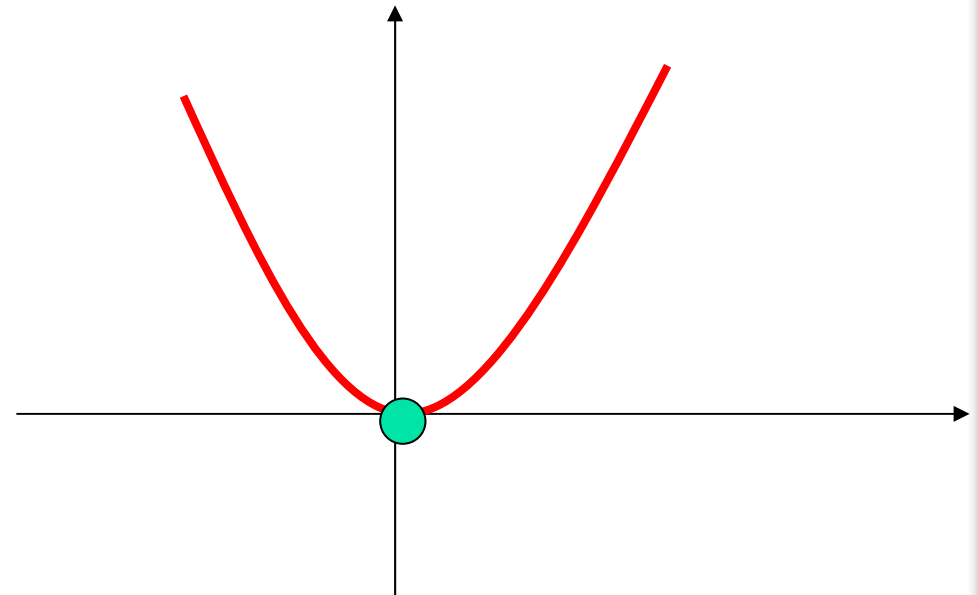
delta	0
x2	0
x1	0
c	0
b	0
a	1

$$x^2 = 0$$

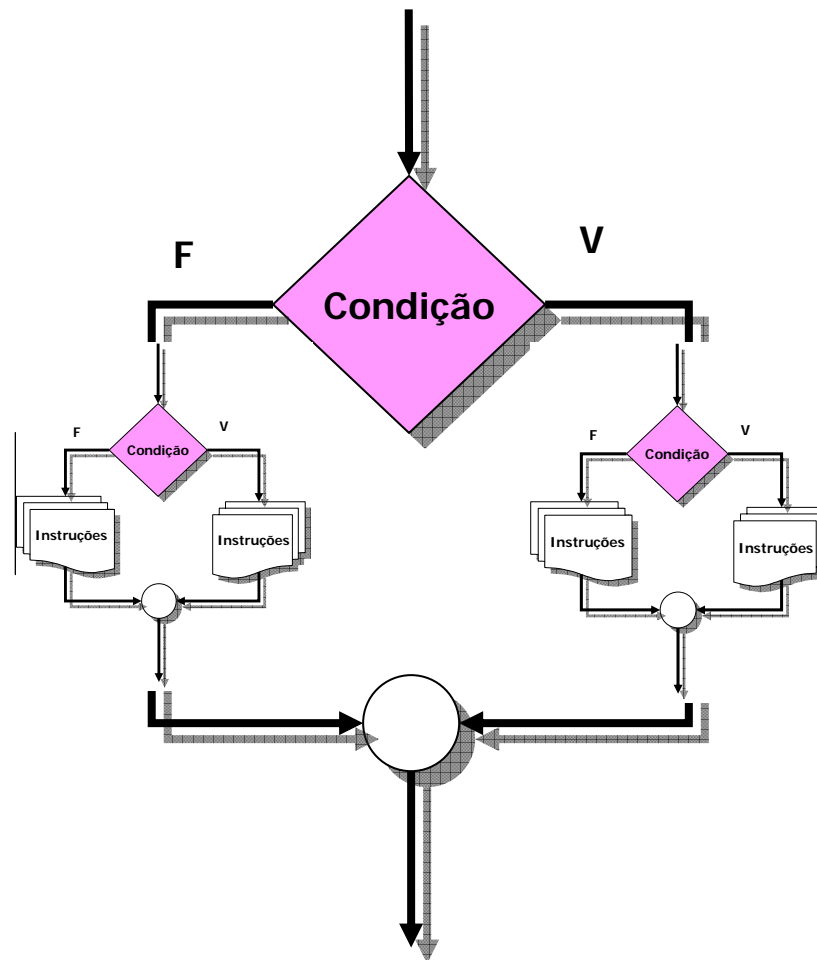
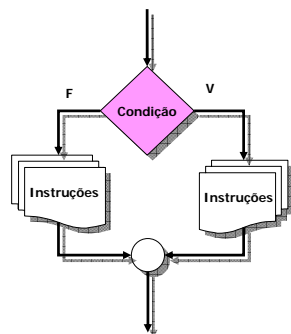
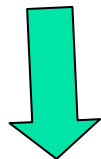
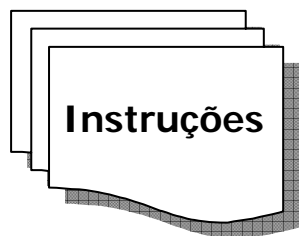
$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$x_1 = 0$$

$$x_2 = 0$$



Seleccção encadeada



Seleção em Alternativa

Linguagem estruturada

Se condicao Entao

Se condicao Entao
intrução

Senão
intrução

Fim_se

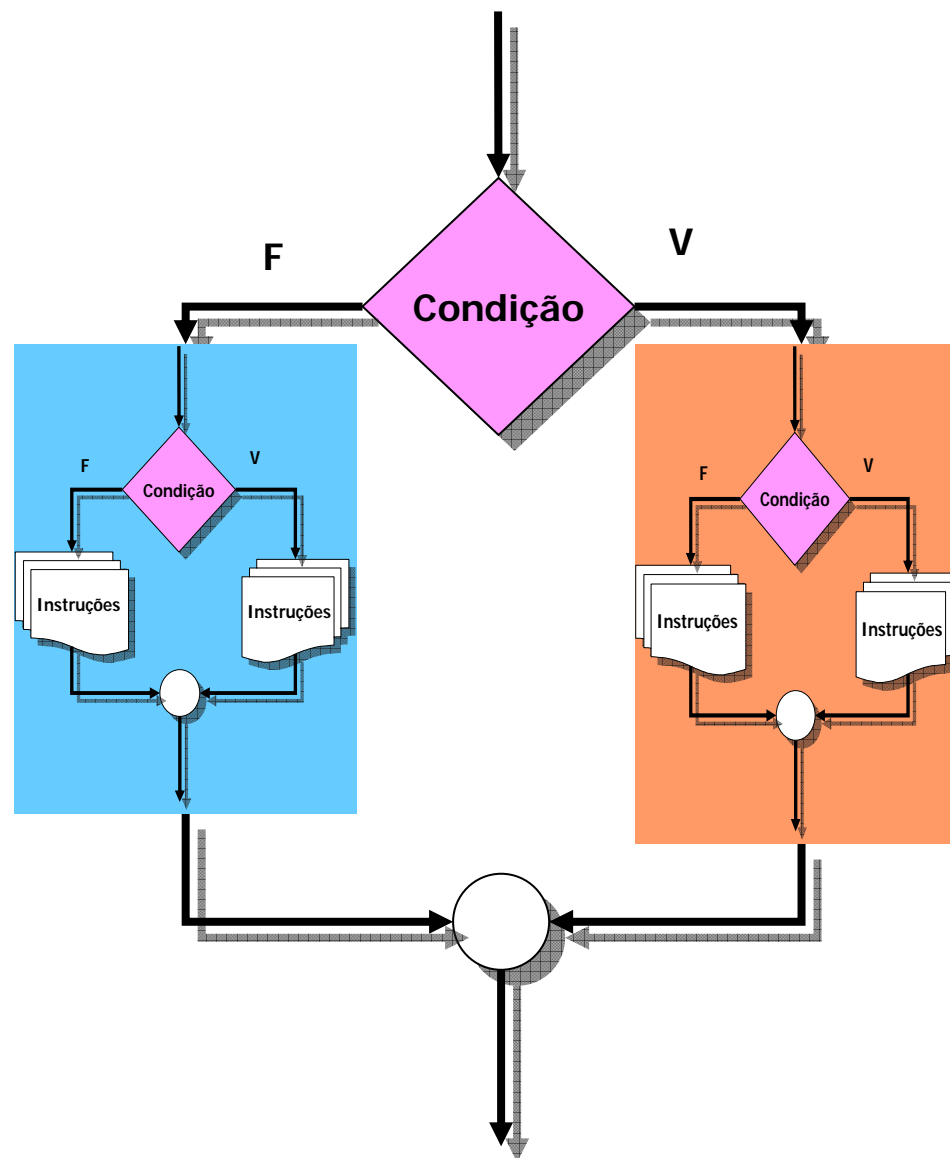
Senão

Se condicao Entao
intrução

Senão
intrução

Fim_se

Fim_se

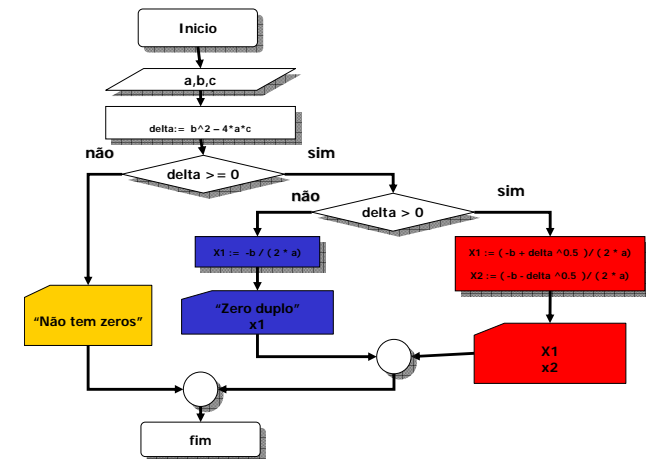


Exercício

Programa Equacao2g

```

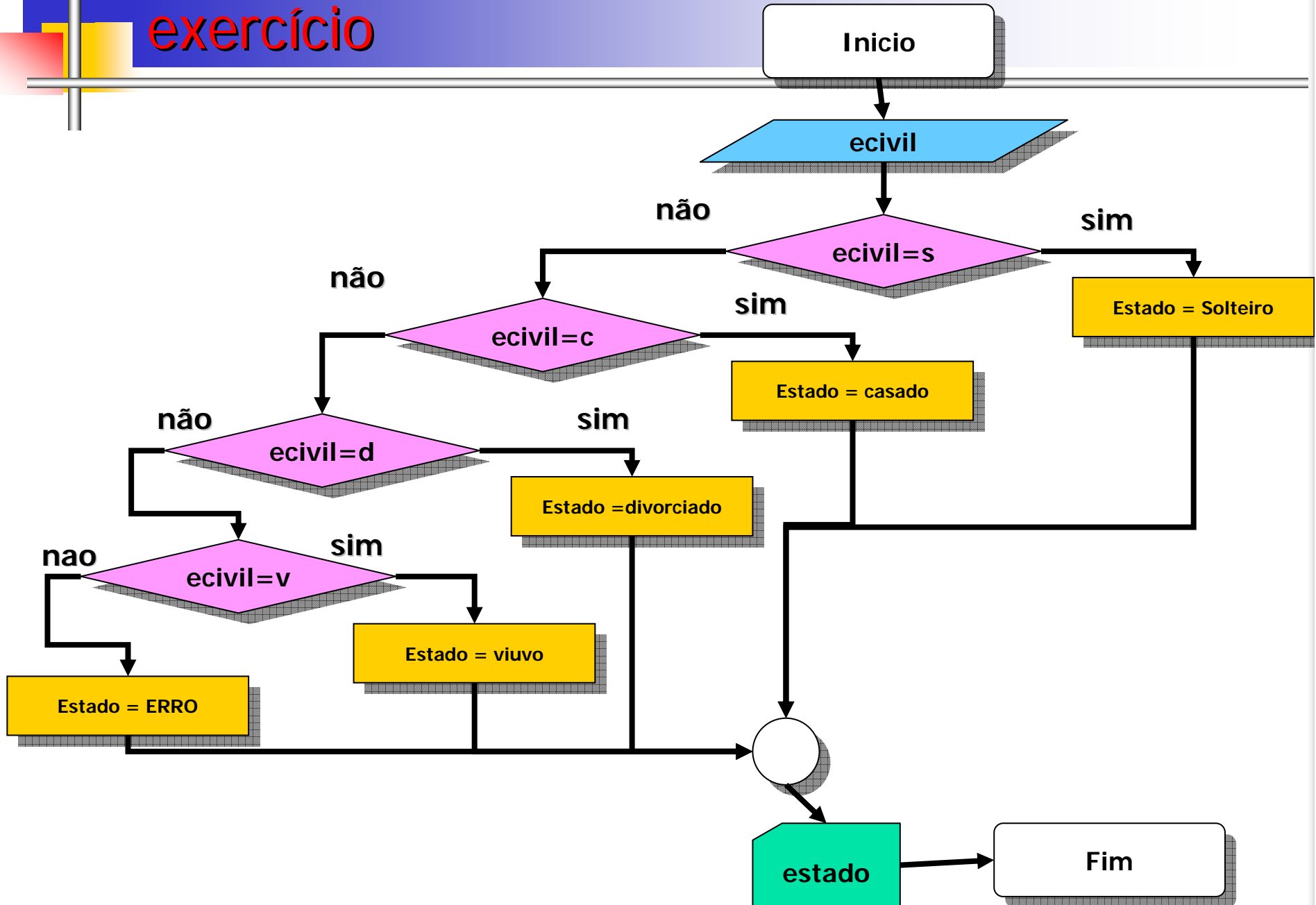
inicio
real a , b , c , x1 , x2 , delta
escrever "Raizes da equação de 2º grau"
escrever "\na = "
ler a
escrever "b = "
ler b
escrever "c = "
ler c
delta <- b ^ 2 - 4 * a * c
se delta >= 0 entao
  se delta = 0 entao
    x1 <- (( b * -1 ) + delta ^ 0.5 ) / ( 2 * a )
    escrever "Zero duplo x1 = " , x1
  senao
    x1 <- (( b * -1 ) - delta ^ 0.5 ) / ( 2 * a )
    x2 <- (( b * -1 ) + delta ^ 0.5 ) / ( 2 * a )
    escrever " x1 = " , x1 , " x2 = " , x2
  fimse
senao
  escrever "não tem raizes"
fimse
fim
  
```



exercício

- Construir um programa que solicite o estado civil de uma pessoa (s, c, d, v) e o imprima por extenso (solteiro , casado, divorciado, viuvo)
 - Entrada
 - Letra
 - Saída
 - Texto
 - Algoritmo
 - ?

exercício

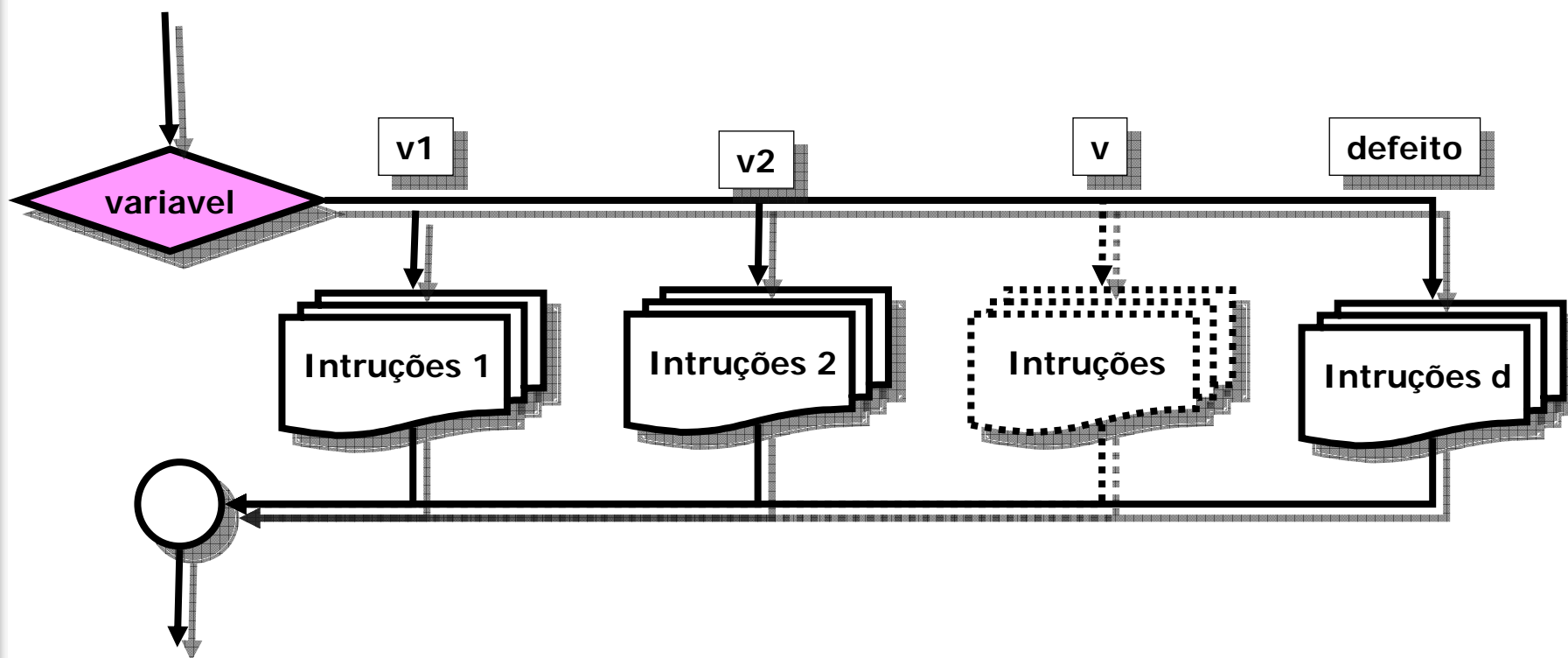




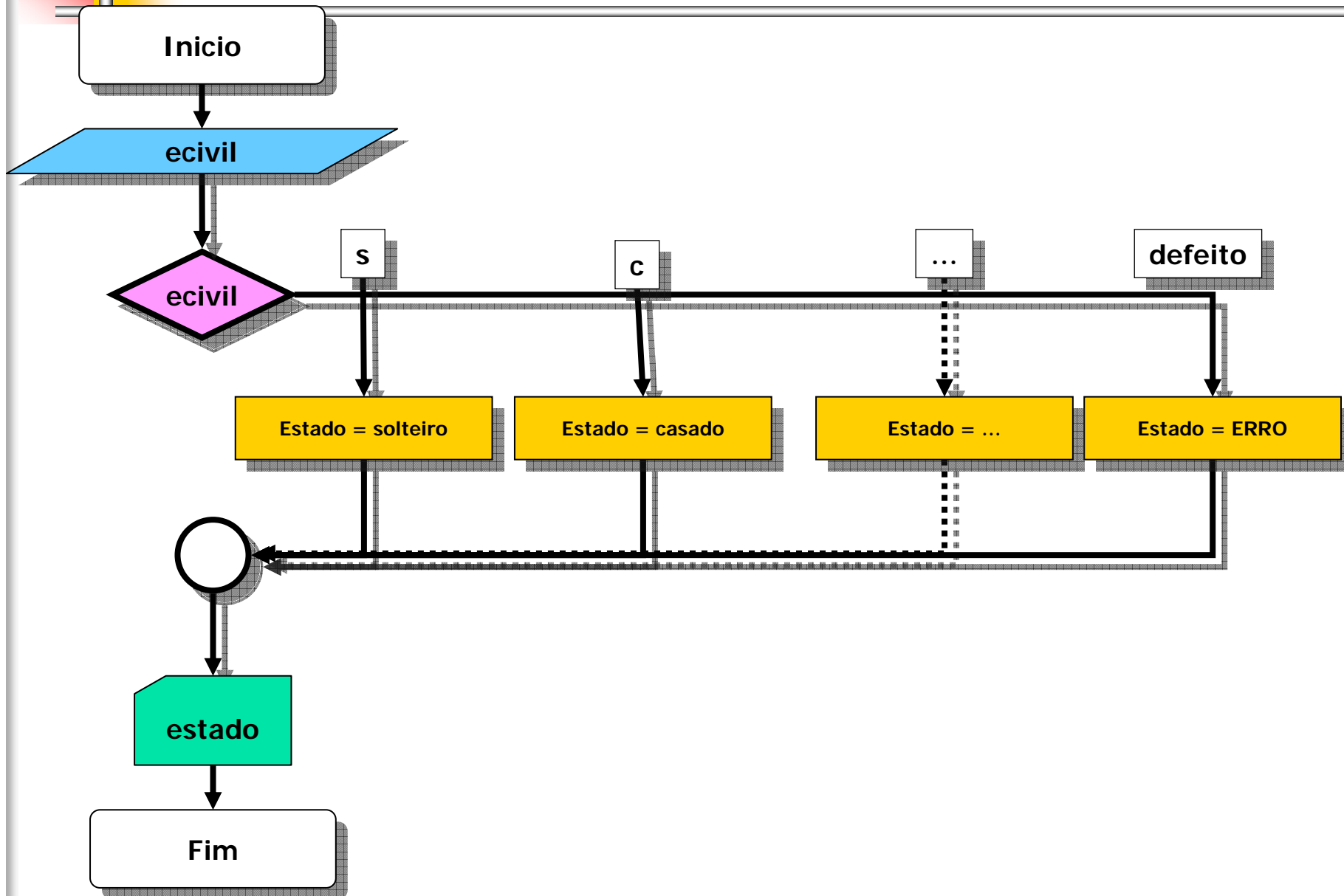
exercício

- Construir um programa que solicite o número do mês e o imprima por extenso
 - Entrada
 - numero
 - Saída
 - Texto
 - Algoritmo
 - ?

Seleccção Múltipla



Seleccção Múltipla



Seleção Múltipla

Linguagem Estruturada

ESCOLHE <EXPRESSAO>

CASO <expressao>:

<bloco-de-instrucoes1...>

CASO <expressao>:

<bloco-de-instrucoes2...>

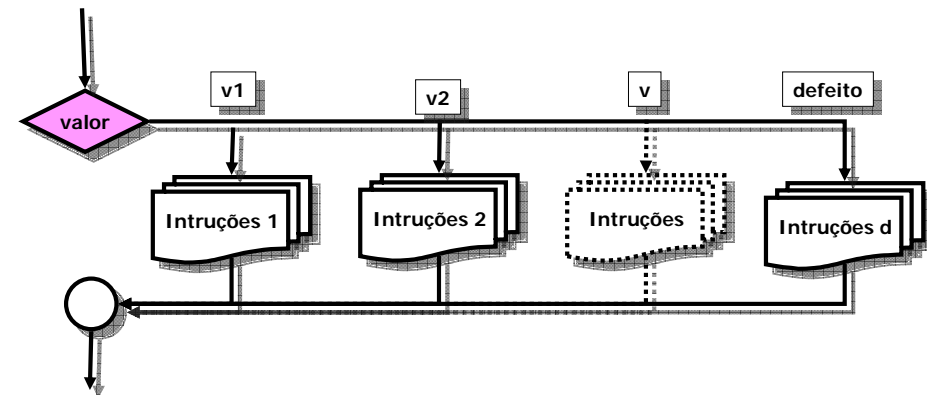
CASO <expressao>:

<bloco-de-instrucoes3...>

defeito :

<bloco-de-instrucoes 4...>

FIMESCOLHE



Exercício

Estado civil

```
inicio
caracter ecivil
escrever "Estado civil (SCDV):"
ler ecivil
escolhe ecivil
  caso "s", "S":
    escrever "solteiro"
  caso "c", "C":
    escrever "Casado"
  caso "d", "Divorçado":
    escrever "solteiro"
  caso "v", "Viuvo":
    escrever "solteiro"
  defeito :
    escrever "erro"
fimescolhe
fim
```




Exercícios

Exercícios

- Pretende-se um programa para calcular a nota final de um aluno de IP ou CPI e imprima se o aluno está aprovado ou reprovado.
- Desenhe o fluxograma
- Construa o programa

Exercícios

- Pretende-se um programa que leia dois números e uma operação (+ , - , * ou /) e imprima o resultado da operação sobre os números.
- Desenhe o fluxograma
- Construa o programa

Exercício

- Pretende-se um programa que dado o número de anos de uma pessoa escreve a sua faixa etária.
- Criança - [0 12]
- Adolescente [13 17]
- Adulto - [18 64]
- idoso - [65 . . .]

Exercícios

Pretende-se imprimir a informação “negativa” “estável” e “positiva” do valor de cotação de uma acção na BVL. O utilizador introduz o valor anterior e o valor actual e o programa imprime a variação

- Desenhe o fluxograma
- Construa o programa

Exercícios

Pretende-se calcular as raízes de um polinómio com grau inferior a 3

- Desenhe o fluxograma
- Construa o programa

$$ax^2 + bx + c$$

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Exercícios

Pretende-se calcular as raízes reais e imaginárias de uma equação de 2º grau.

- Desenhe o fluxograma
- Construa o programa

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$i = \sqrt{-1}$$

Exercícios

Pretende-se solicitar dois números ao utilizador e o programa troca os valores das variáveis.

- Desenhe o fluxograma
- Construa o programa

X1 = 20

X2 = 30

X1 = 30

X2 = 20

Exercícios

Pretende-se solicitar dois números ao utilizador e imprimê-los por ordem crescente.

- Desenhe o fluxograma
- Construa o programa

X1 = 50

X2 = 30

X1 = 30

X2 = 50

Exercício

- Construa um programa que imprima se um ano é bissexto ou comum
 - Todos os múltiplos de 400
 - Todos os que são múltiplos de 4 e não são múltiplos de 100

- 2000
 - Bissexto
- 2001
 - Comum
- 2100
 - Comum
- 2004
 - bissexto

Exercício

- Construa um programa que solicite ao utilizador uma data e imprima o dia seguinte.

Exercício

- Construa um programa que solicite a data de nascimento de um aluno e verifique se a data é válida ou não
- Construa um programa que solicite o nome e a data de nascimento de dois alunos e imprima o nome do mais velho.