

What is an algorithm? Credit: Bob Al-Greene / Mashable

<https://mashable.com/article/what-is-an-algorithm>

Algoritmo

Conjunto finito de instruções, bem definidas e não ambíguas que, ao serem executadas numa determinada ordem, resolvem um problema.

Um algoritmo deve ser:

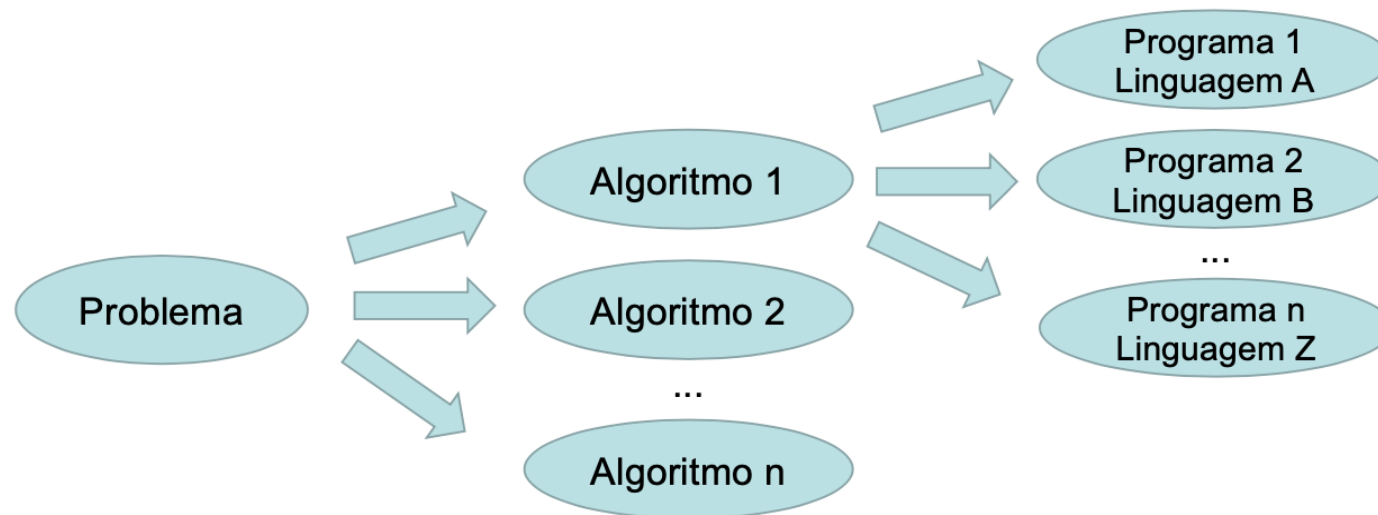
- Genérico;
- Determinístico;
- Finito;
- Eficaz;
- Capaz de comunicar.


$$\begin{array}{r} 1 \ 1 \\ 345 \\ + 278 \\ \hline 623 \end{array}$$

algoritmo para somar dois números inteiros
(<https://www.scribbr.com/ai-tools/what-is-an-algorithm/>)

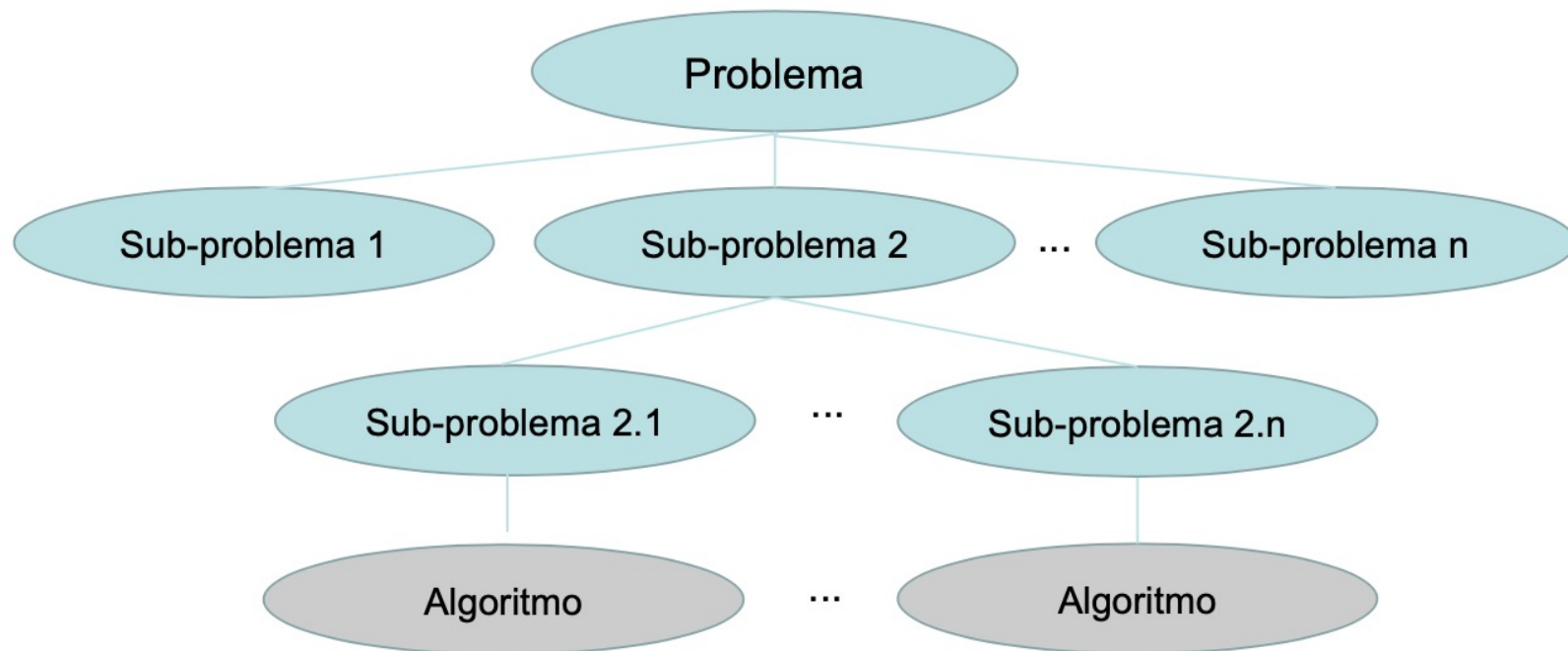
Algoritmo

- Para o mesmo problema podem existir vários algoritmos diferentes que o resolvem.
- Cada algoritmo pode dar origem a diferentes programas, cada um deles codificado numa linguagem de programação.



Decomposição Algorítmica (abordagem Top-Down)

Problemas complexos devem ser divididos em subproblemas sucessivamente mais simples – “dividir para conquistar”.



Representação de Algoritmos

- **Linguagem corrente:** menos adequada para a programação de computadores.
- **Fluxogramas** (ou diagramas de fluxo; *flowcharts*): podem ser de leitura difícil para problemas complexos.
- **Linguagem algorítmica** (pseudo-código): a representação mais próxima das linguagens de programação.



- Plataforma web que usa a linguagem algorítmica Portugal

Linguagem Corrente

Instruções de utilização de um shampoo:

- 1. Molhar a cabeça abundantemente**
- 2. Colocar o shampoo na cabeça e esfregar**
- 3. Repetir a partir do passo 1**

Será um bom algoritmo?

Linguagem Corrente

Representar em linguagem corrente um algoritmo que indique se um determinado número inteiro é par ou ímpar.

1. **Ler um número inteiro, n**
2. **Calcular o resto da divisão de n por 2 e guardar em r**
3. **Se r for igual a 0, então escreve que é número par**
4. **Senão, escreve que é número ímpar**

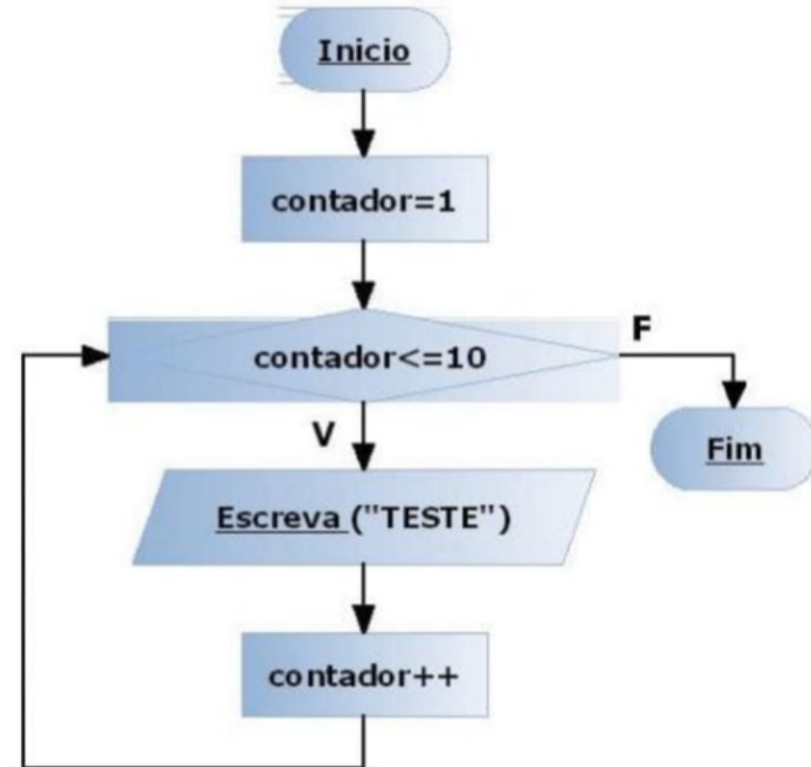
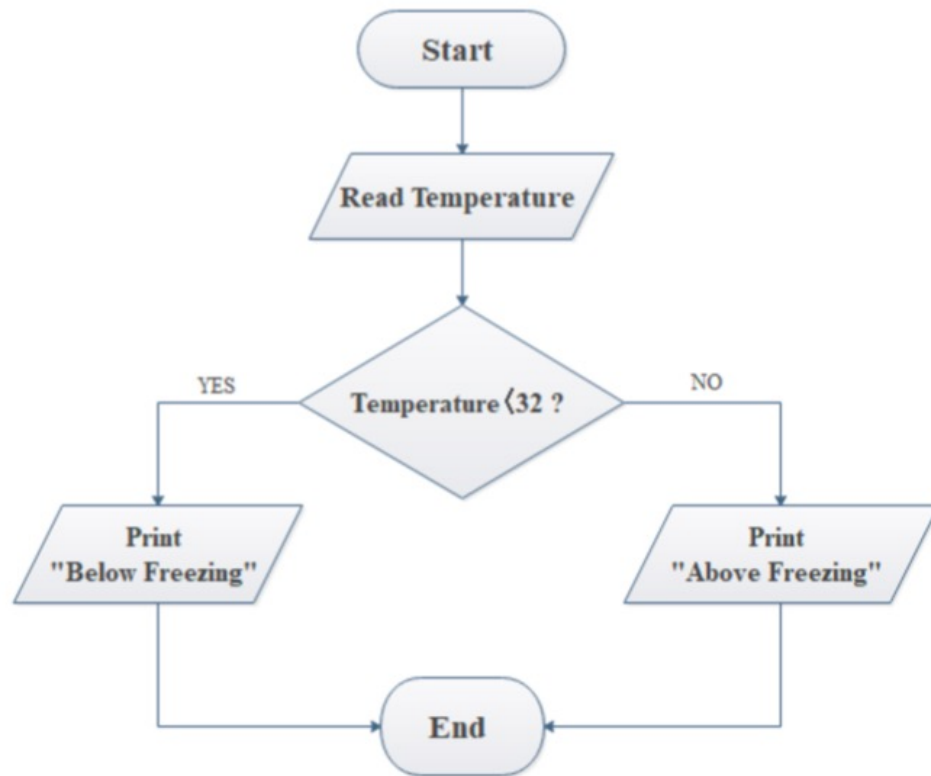
As letras n e r representam **variáveis**. Variáveis são entidades abstratas usadas no algoritmo.

Fluxogramas

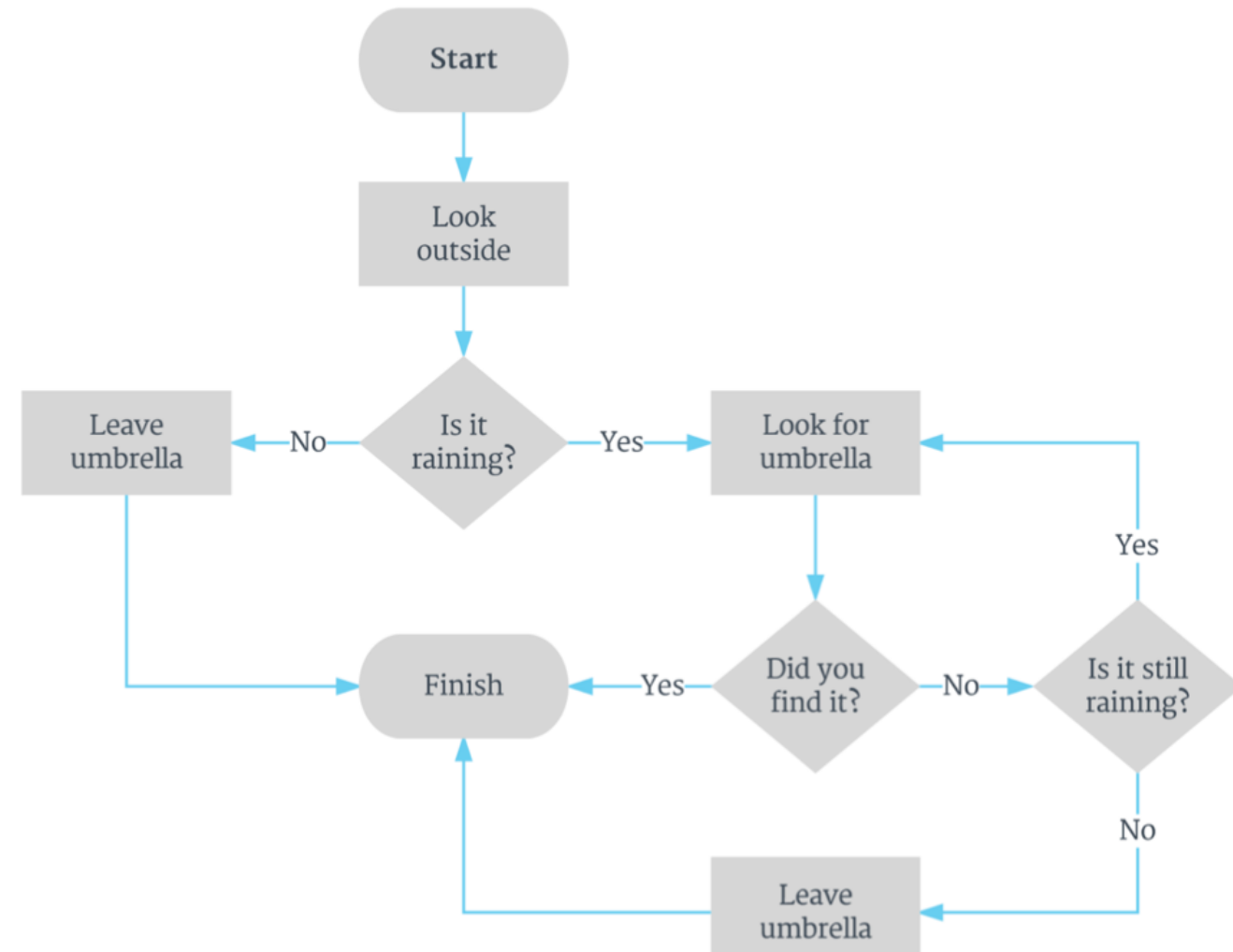
Representação do algoritmo sob a forma de um esquema que reflete a sequência de passos para resolver o problema (diagrama de fluxo).

- Existem diferentes formas de escrever fluxogramas;
- Aplicam-se na programação de computadores e em muitas outras áreas;
- Os símbolos usados variam de acordo com as áreas de aplicação e as abordagens seguidas.

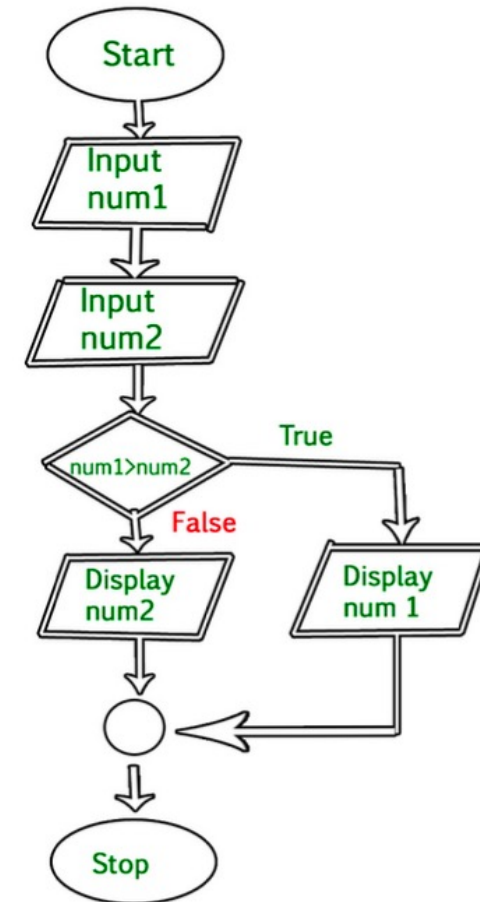
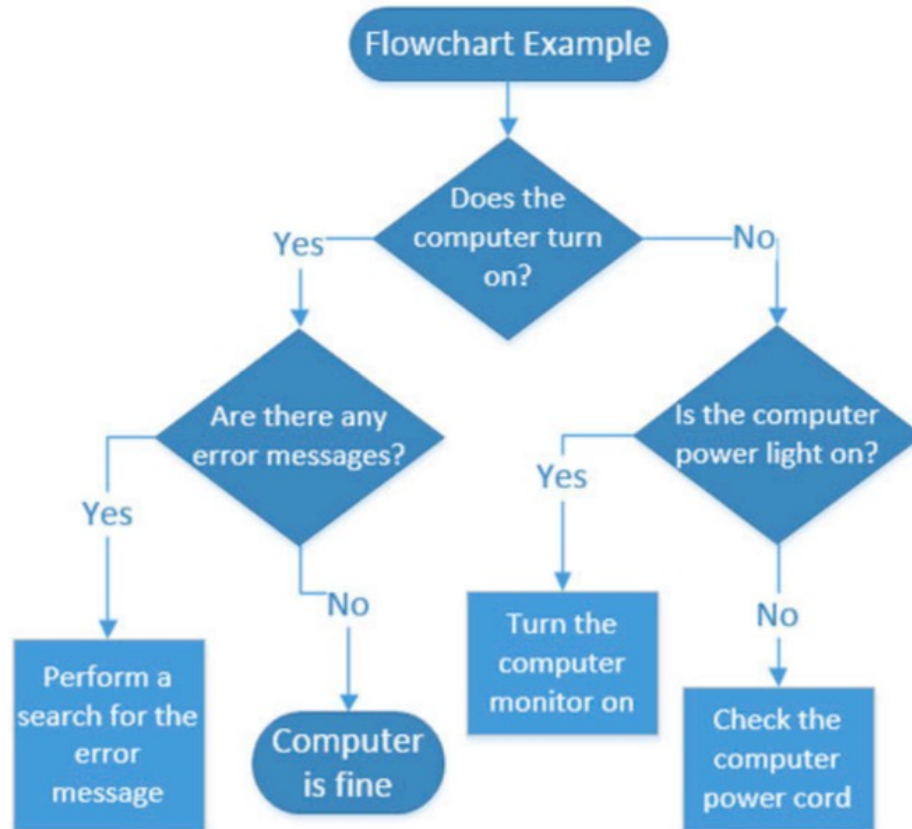
Fluxogramas



Fluxogramas



Fluxogramas





Início do fluxograma



Fim do fluxograma



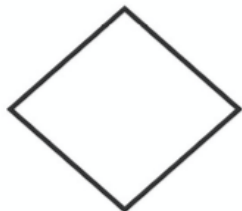
Execução de uma ação



Leitura de dados

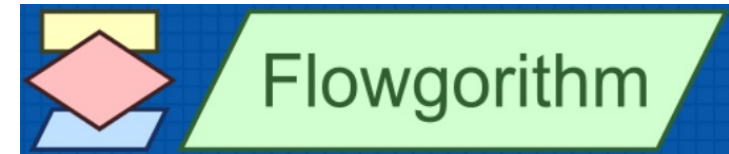


Escrita de resultados

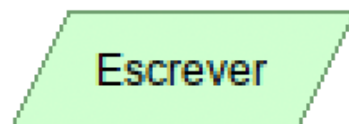
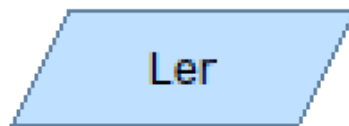


Teste de condição

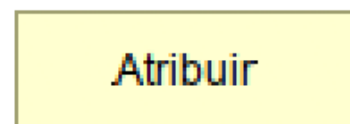
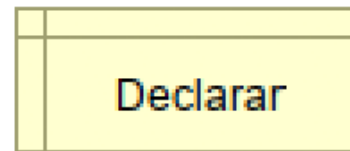
Símbolos para Fluxogramas



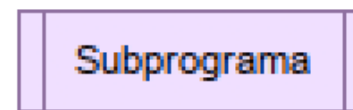
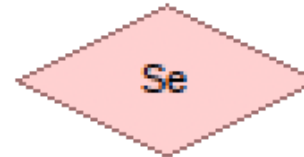
Entrada / Saída



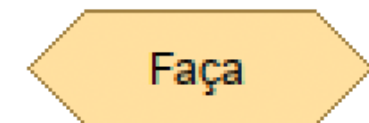
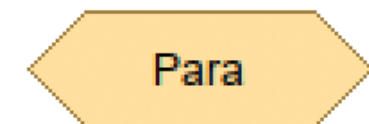
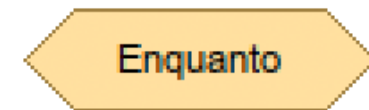
Variáveis



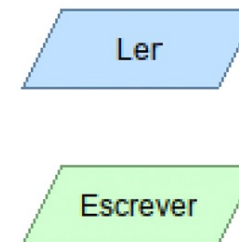
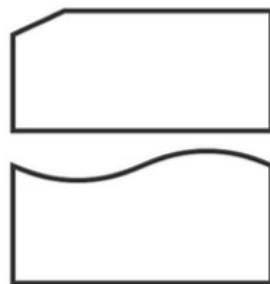
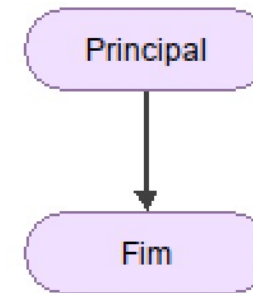
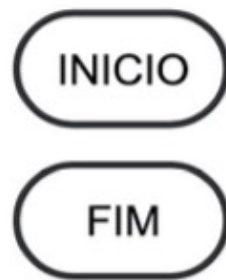
Controlo



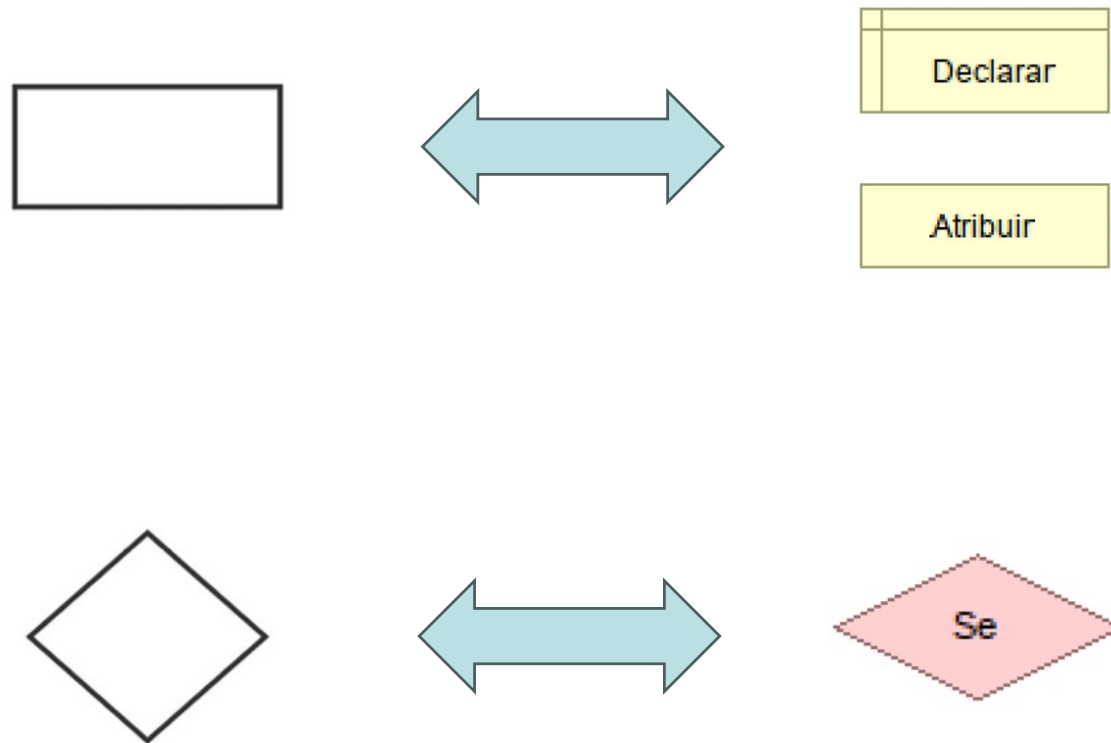
Ciclos de Repetição



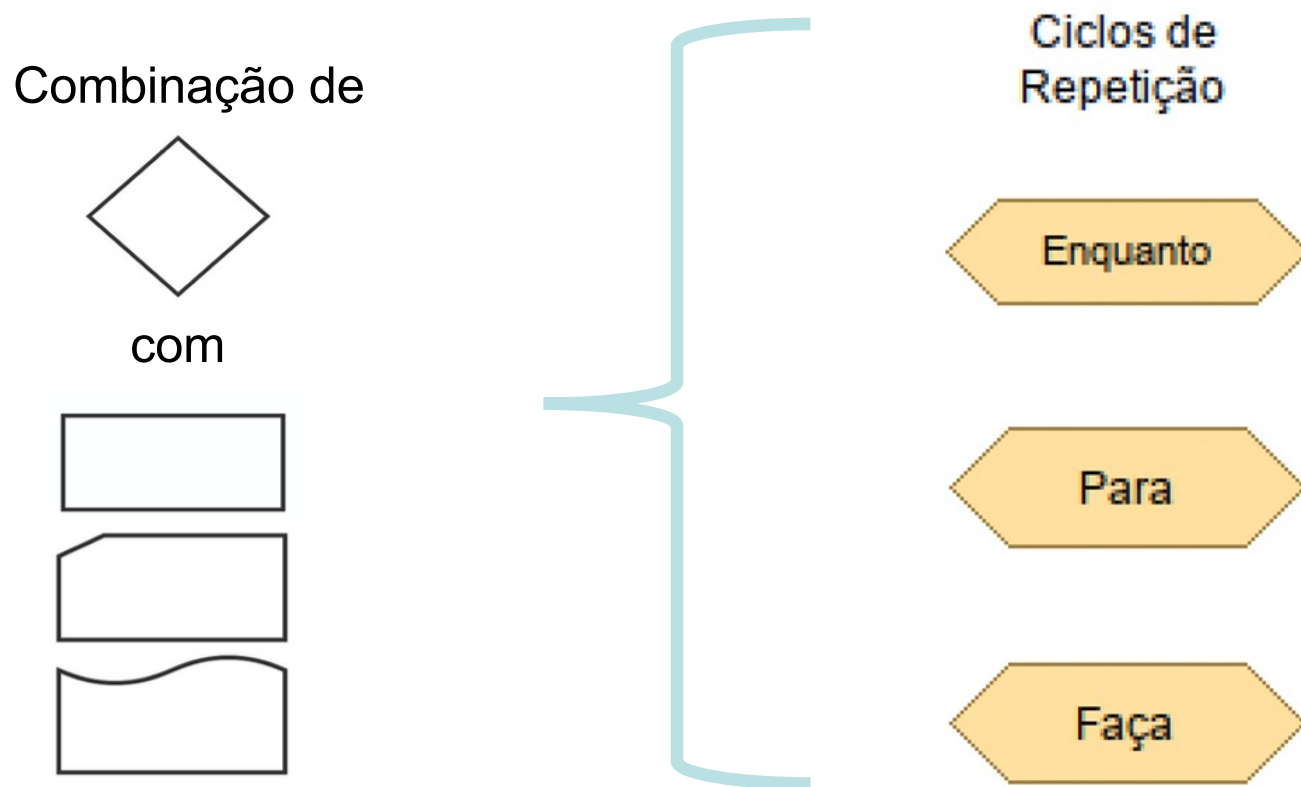
Equivalências de Símbolos

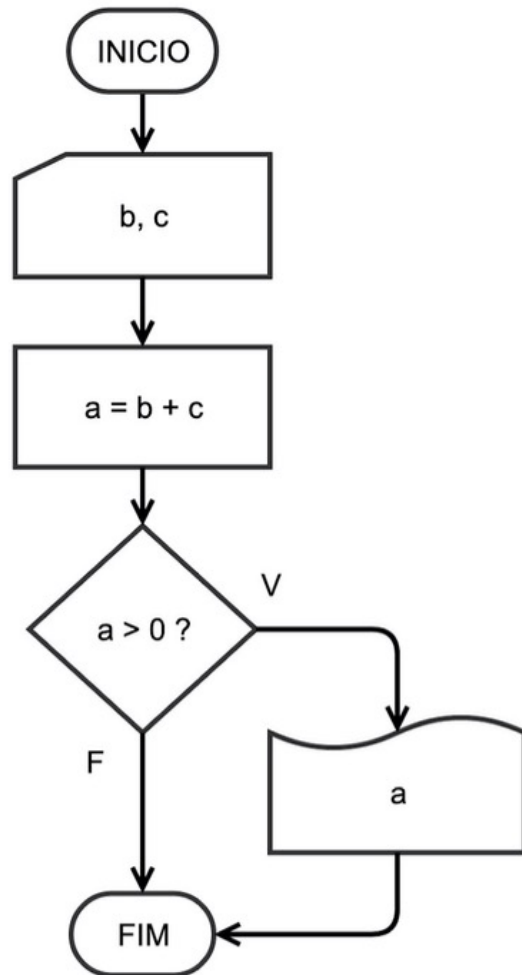


Equivalências de Símbolos



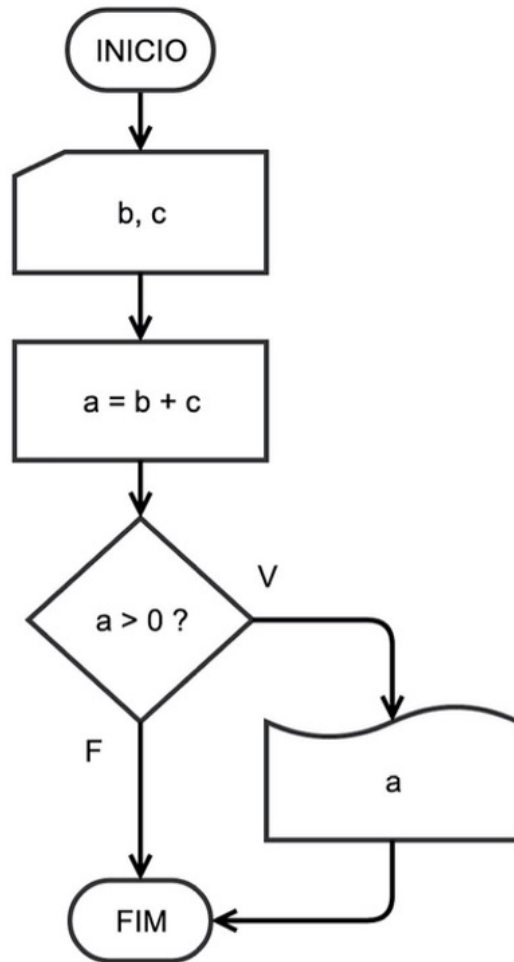
Equivalências de Símbolos



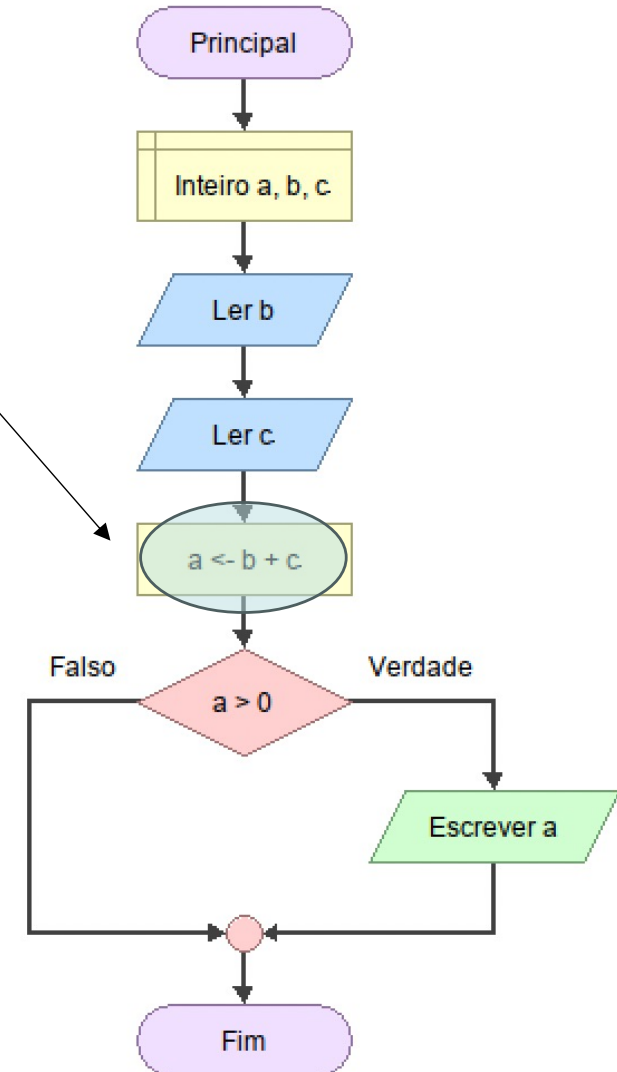


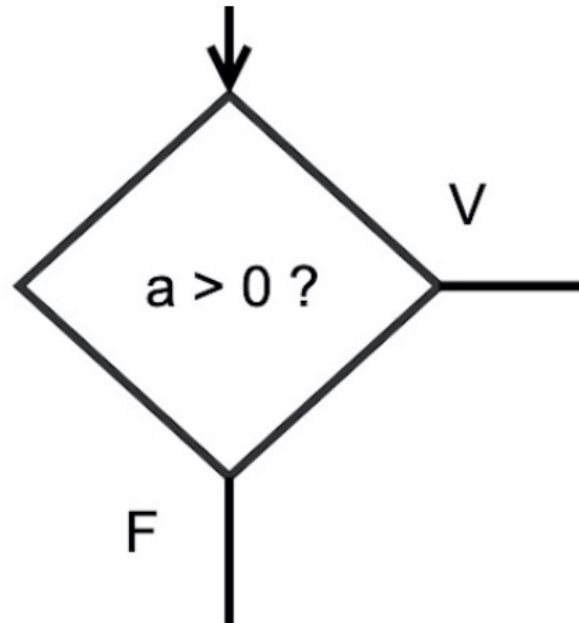
1. Ler dois números inteiros, b e c.
2. Somar b com c e guardar em a.
3. Se a for maior do que 0, então escreve o valor de a e termina.
4. Senão, não escreve nada e termina.

Os símbolos usados no fluxograma são interligados por linhas com setas para indicar o sentido do fluxo do diagrama, ou seja, permitem indicar a sequência dos vários passos do algoritmo.



$a \leftarrow b + c$
 “ \leftarrow ” equivale a “=”





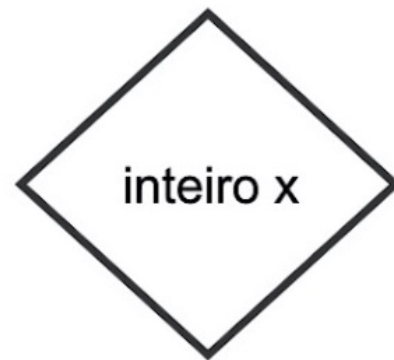
Teste de Condição:

- O fluxo do algoritmo entra por um dos vértices;
- A condição a testar só pode ter como resultado “verdadeiro” ou “falso”;
- O fluxo do algoritmo sai por dois dos outros vértices: uma saída se a condição for verdadeira e outra saída se a condição for falsa.

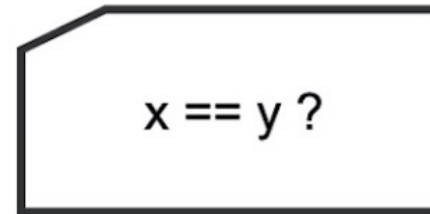
Exemplo de expressões a usar como condições:

$a > b$
 $x == y$
 $a < b$ e $a < c$
 $r > 0$

Estes blocos estarão corretos?



Teste de condição



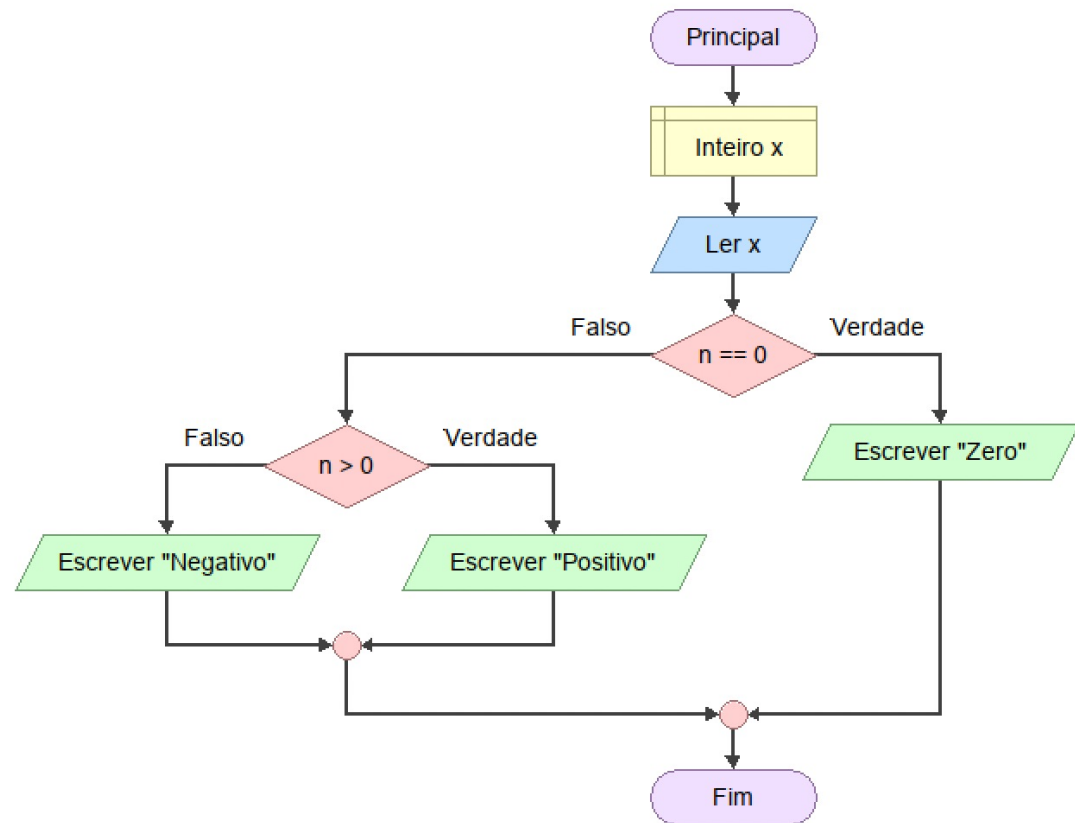
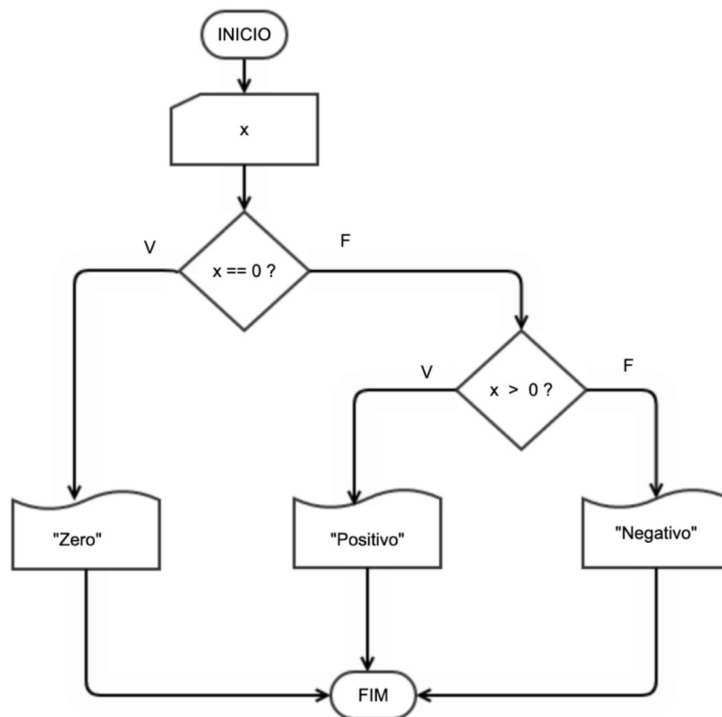
Leitura de dados

Exercício: Algoritmo para verificar se um número inteiro é positivo, negativo ou zero

1. Ler um número inteiro
2. Comparar o número com 0
3. Se o número for igual a 0, então escrever "zero"
4. Se o número for menor que zero, então escrever "negativo"
5. Senão, escrever "positivo"

Representar o algoritmo anterior através de um fluxograma.

Exercício: Algoritmo para verificar se um número inteiro é positivo, negativo ou zero

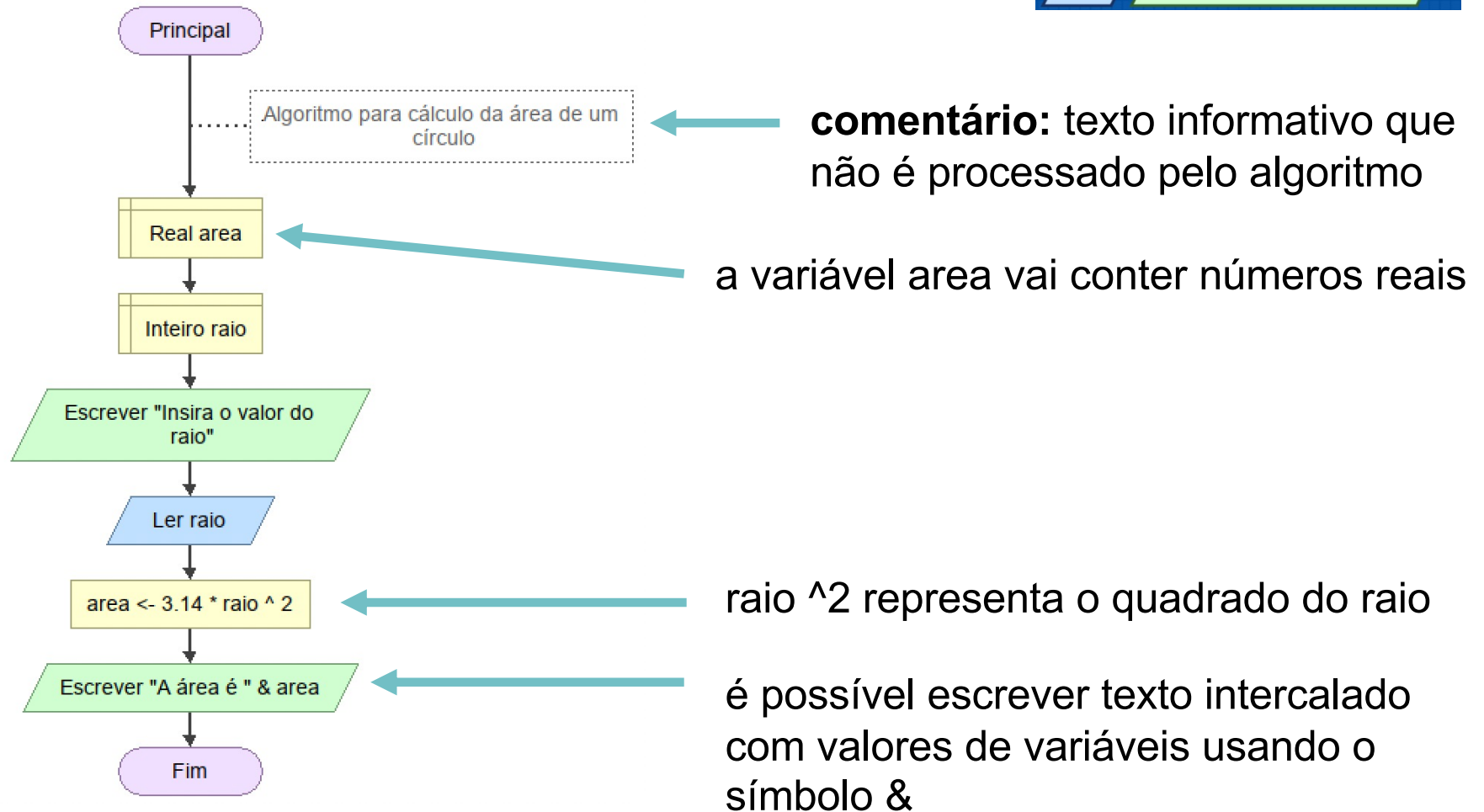
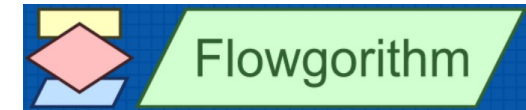


Exercício: Calcular a área de um círculo

1. Ler o raio do círculo, r
2. Calcular $area = \pi \times r^2$
3. Escrever $area$
4. Fim

Representar o algoritmo anterior através de um fluxograma, considerando que o círculo tem um raio que é um número inteiro.

Exercício: Calcular a área de um círculo.

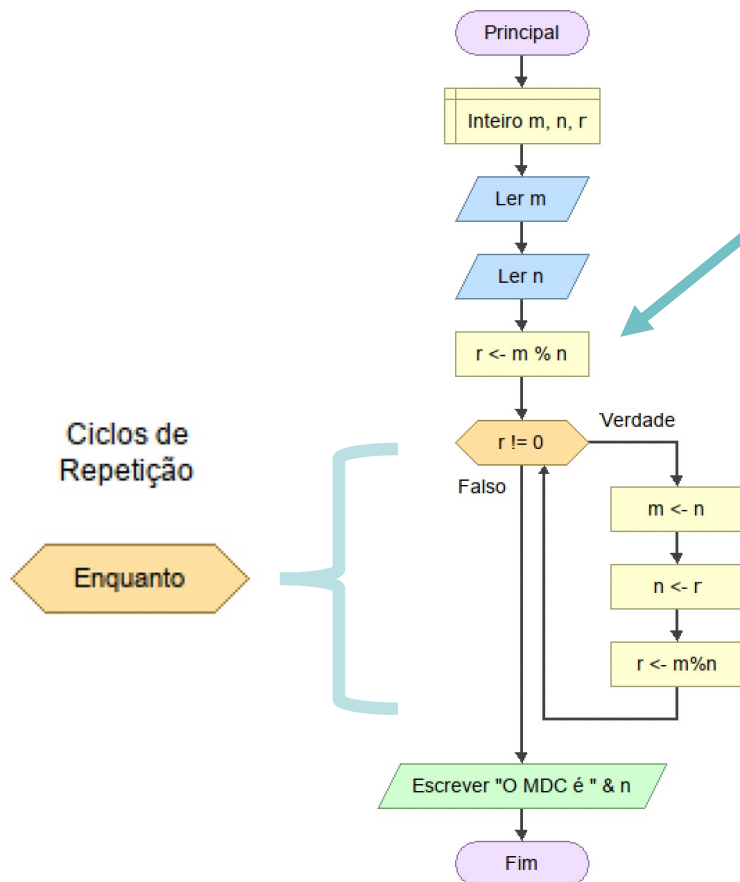
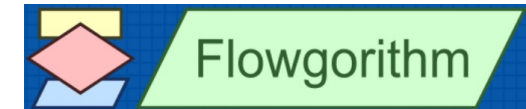


Exercício: Calcular o máximo divisor comum (MDC) de dois números inteiros

1. Ler dois números inteiros, m e n
2. Fazer a divisão inteira de m por n e guardar o resultado em r
3. Se r for igual a 0, vai para o passo 7
4. Atribuir a m o valor de n
5. Atribuir a n o valor de r
6. Voltar ao passo 2
7. O MDC de m e n é n
8. Fim

Representar o algoritmo anterior através de um fluxograma.

Exercício: Calcular o máximo divisor comum (MDC) de dois números inteiros



% representa o “resto da divisão”

!= significa “diferente de” (ou *não igual*)

2. Fazer a divisão inteira de m por n e guardar o resultado em r
3. Se r for igual a 0, vai para o passo 7
4. Atribuir a m o valor de n
5. Atribuir a n o valor de r
6. Voltar ao passo 2

Problema

De um conjunto de 25 moedas de ouro, sabe-se que uma é falsa e pesa menos do que as restantes. Existe disponível uma balança de pratos que, no entanto, apenas pode efetuar um máximo de 3 pesagens. Descubra um algoritmo que permita encontrar a moeda falsa usando a balança de pratos.

Ver solução em:

<https://www.quora.com/Suppose-that-we-have-25-coins-that-look-identical-The-coins-are-all-the-same-except-that-one-coin-is-counterfeit-and-heavier-than-the-others-How-can-one-determine-in-three-weighings-on-a-balanced-scale-which-coin-is>

Torres de Hanoi: O problema das Torres de Hanoi (a capital do Vietname, que dá o nome ao jogo) consiste em considerar 4 discos de tamanhos diferentes empilhados numa haste vertical (do menor para o maior). O objectivo é mover os discos de uma haste para outra com o auxílio de uma terceira de forma a que nunca seja colocado um disco maior sobre um menor. Este problema foi pela primeira vez colocado pelo matemático francês Edouard Lucas, em 1883, com 8 discos.



<https://www.mathsisfun.com/games/towerofhanoi.html>