

Unidade 2: Algoritmos e programação



2.1. Algoritmos

Os computadores estão sendo utilizados para resolver problemas cada vez de maior porte e complexidade. A utilização do computador para resolver problemas é precedida pela necessidade de se desenvolver um algoritmo.

Um algoritmo pode ser definido como sendo um conjunto de instruções (não ambíguas e finitas) que são executadas até que determinado objetivo seja atingido. A Figura 2.1 ilustra de forma simplificada a estrutura de um algoritmo.

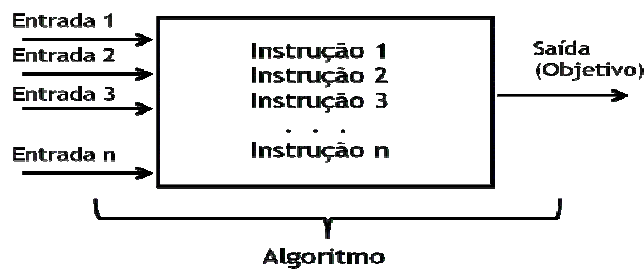


Figura 2.1: Estrutura simplificada de um algoritmo.

Um exemplo de um algoritmo seria os passos descritos em uma receita culinária para elaboração de uma torta ou bolo. Vejamos como ficaria a Figura 2.1 no caso do preparo de um bolo. As entradas seriam os ingredientes necessários, por exemplo, farinha, açúcar, sal, leite, ovos, fermento, manteiga. As instruções são as etapas descritas na receita, ou seja, o modo de preparo: (1) bata a manteiga até ficar cremosa; (2) adicione o açúcar e as gemas; (3) bata até formar um creme; (4) adicione a farinha, o fermento, o sal e o leite; (5) misture bem; transfira para uma assadeira; (6) leve ao forno por 45 minutos. A saída será o bolo pronto.

A correta execução dos passos contidos na receita, contudo, podem não garantir a obtenção do resultado esperado (o fermento utilizado pode não estar bom, as quantidades adicionadas dos componentes não foram corretas, por exemplo).

Além do mais um algoritmo corretamente executado não irá solucionar um problema se estiver implementado de forma incorreta ou ainda se não for apropriado ao problema. No caso do preparo do bolo, as instruções precisam ser seguidas em uma ordem pré-definida. Não há como levar a mistura ao forno sem antes prepará-la. Em alguns casos, a inversão na ordem de algumas instruções poderá dificultar a solução do problema, ou mesmo levar a algum resultado não esperado (incorreto ou inconsistente).

Vejamos mais um exemplo simples. Imagine um algoritmo para uma pessoa se vestir. As entradas poderiam ser: meias, calça, camisa e sapato. As instruções seriam: (a) vestir meias; (b) vestir sapato; (c) vestir calça; (d) vestir camisa. O objetivo final é a pessoa vestida. Nesse caso, independente da ordem das instruções, o objetivo sempre será alcançado. Contudo, a depender da seqüência das instruções o resultado será diferente em termos da dificuldade para realização e do tempo de solução. Compare as estas diferentes seqüências de instruções:

- (1) (a) vestir meias; (b) vestir sapato; (c) vestir calça; (d) vestir camisa;
- (2) (d) vestir camisa; (c) vestir calça; (a) vestir meias; (b) vestir sapato;
- (3) (a) vestir meias; (d) vestir camisa; (c) vestir calça; (b) vestir sapato.

Podemos chegar à conclusão que as três propostas apresentam diferentes graus de dificuldade (vestir a calça tendo o sapato já calçado é mais difícil que vestir a calça sem ter ainda os sapatos calçados). Contudo, o resultado final será o mesmo.

Todo algoritmo deveria apresentar cinco características importantes:

- **Entradas:** o algoritmo precisa possuir entradas, ou seja, receber as informações que são necessárias para sua execução.
- **Saídas:** o algoritmo deve ter uma ou mais saídas, que é (são) o(s) objetivo(s) atingido(s).
- **Finitude:** o algoritmo deve terminar após um número finito de passos.
- **Instruções elementares:** cada instrução deve ser claramente definida (sem ambigüidades).
- **Efetividade:** as instruções devem, em princípio, ser executadas com precisão empregando papel e lápis.



2.1.1. Como representar os algoritmos ?

Existem várias formas de representação para os algoritmos. Vejamos alguns exemplos:

- **Linguagem natural ou algoritmo informal:** os algoritmos são expressos diretamente na linguagem escrita. Veja o exemplo de algoritmo para substituição de uma lâmpada queimada.

Entradas: escada, lâmpada

Instruções: (1) posicionar a escada embaixo da lâmpada;
(2) subir na escada até alcançar a lâmpada;
(3) remover a lâmpada queimada do bocal;

(4) rosquear a nova lâmpada no bocal;

(5) descer da escada;

Saída (objetivo): lâmpada substituída, iluminação re-estabelecida.

- **Fluxograma convencional ou Fluxograma:** os algoritmos são representados empregando formas geométricas padronizadas para indicar as diferentes instruções e decisões que devem ser executadas para resolver o problema. O fluxograma foi uma das primeiras ferramentas utilizadas para representar algoritmos. Veremos exemplo mais adiante.
- **Pseudo-código:** esta forma de representação emprega uma linguagem intermediária entre a linguagem natural e a linguagem de programação para representar os algoritmos. São constituídas por um vocabulário de uma linguagem natural (português, inglês, etc) e pela sintaxe de uma linguagem de programação. Quando os termos empregados estão em Português alguns autores o chamam de **Portugol**.







Existem ainda outras ferramentas utilizadas para representar os algoritmos. Dentre elas podemos citar os diagramas de Nassi-Shneiderman (cartas N-S), as tabelas de decisão, os diagramas de ação, os diagramas de Warnier, os diagramas de Jackson, por exemplo.



2.1.2. Fluxogramas

A idéia básica desta forma de representação dos algoritmos é empregar figuras geométricas na representação de cada instrução (ou passo) que compõe o algoritmo. Existe um conjunto específico de símbolos empregados na elaboração dos fluxogramas. Estes símbolos são definidos pela norma ISO 5807 (ISO, International Organization for Standardization). Apresentam-se na Tabela 2.1 os mais importantes.

Tabela 2.1: Simbologia gráfica empregada na elaboração de fluxogramas. Fonte: Adaptado da norma ISO 5807		
Símbolo	Nome	Finalidade
	Terminal	Utilizado para representar o início e o fim do fluxo lógico de um programa. Empregado também na definição de sub-rotinas de procedimento ou função.
	Entrada manual	Utilizado para representar a entrada manual de dados, via de regra pelo teclado do computador.
	Processamento	Utilizado para representar a execução de uma operação ou grupo de instruções que estabelecem o resultado de uma operação

		lógica ou matemática.
	Exibição	Utilizado para representar a execução da operação de saída visual de dados em um monitor de vídeo conectado ao computador.
	Documento	Utilizado para representar a execução da operação de saída de dados em um documento emitido por uma impressora na forma de relatório.
	Decisão	Utilizado para representar o uso de desvios condicionais para outros pontos do programa de acordo com situações variáveis.
	Conector	Utilizado para representar a entrada ou saída em outra parte do diagrama de blocos. Pode ser usado na definição de quebras de linha e na continuação da execução de decisões.
	Processo pré-definido	Utilizado para representar um grupo de operações estabelecidas como uma subrotina de processamento anexa ao diagrama de blocos (referência a um subprograma externo).
	Linha (com seta)	Utilizado para representar o vínculo existente entre os vários símbolos de um diagrama de blocos. Indica o sentido de fluxo de execução.



2.1.3. Exemplos de fluxogramas

Apresentaremos alguns exemplos de fluxogramas implementados utilizando a norma ISO 5807.

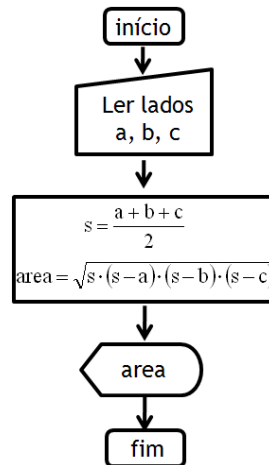
Exemplo 1: Elaborar algoritmo para calcular a área de um triângulo (area) empregando a fórmula de Herão (ou de Heron). São fornecidos os comprimentos dos três lados do triângulo (a, b, c).

$$\text{Fórmula para cálculo da área: } \text{area} = \sqrt{s \cdot (s - a) \cdot (s - b) \cdot (s - c)}$$

onde s é o semi-perímetro do triângulo, calculado por: $s = \frac{(a + b + c)}{2}$

* Observe que a ordem de cálculo é muito importante para resolução deste problema. Primeiro é preciso calcular o perímetro (s) e em seguida a área (area).

Solução:



Exemplo 2: Elaborar algoritmo para o cálculo da média final (MF) dos alunos em uma disciplina do curso de Engenharia Ambiental. Nesta disciplina os alunos realizaram duas provas (P1 e P2) e um trabalho (T), e o seguinte critério de avaliação foi estabelecido pelo professor:

P1 - nota da primeira avaliação

P2 - nota da segunda avaliação

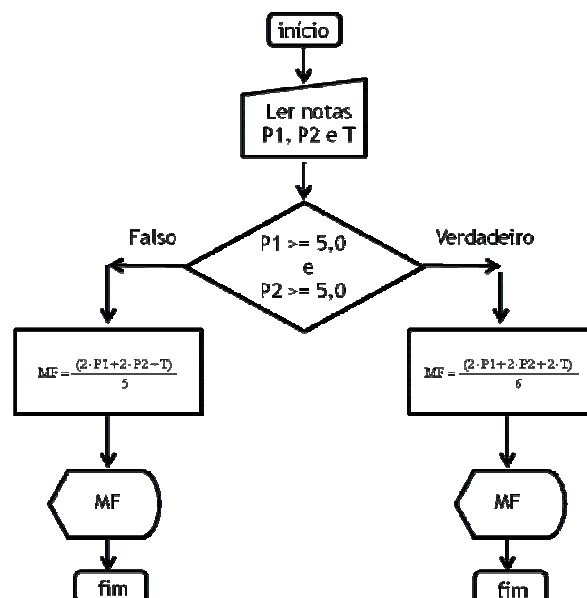
T - nota do trabalho realizado em equipe

MF - média final

Se $P1 \geq 5,0$ e $P2 \geq 5,0$ a média final é calculada por: $MF = \frac{(2 \cdot P1 + 2 \cdot P2 + 2 \cdot T)}{6}$

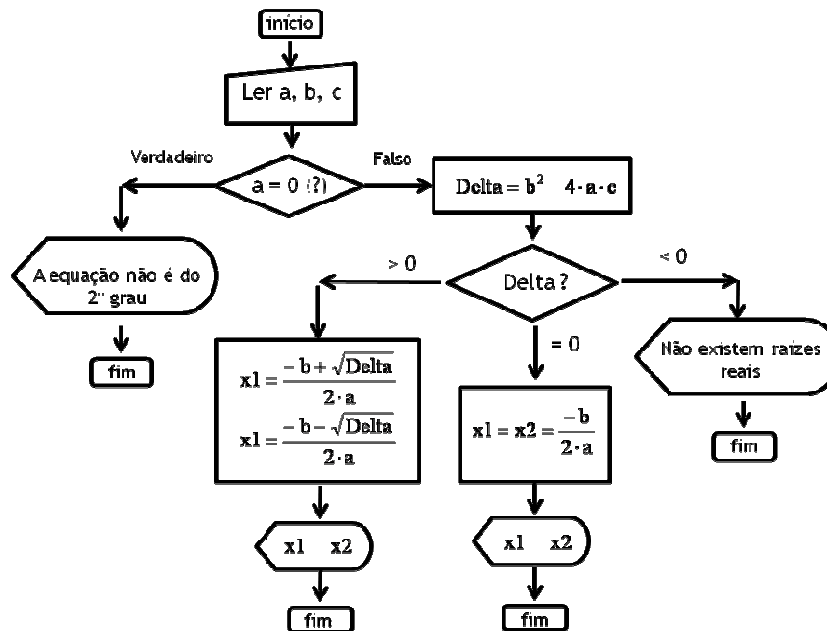
Caso contrário: $MF = \frac{(2 \cdot P1 + 2 \cdot P2 + T)}{5}$

Solução:



Exemplo 3: Formule algoritmo para calcular as raízes (x_1 e x_2) de uma equação do segundo grau ($a \cdot x^2 + b \cdot x + c = 0$) a partir dos valores das constantes a , b e c fornecidas pelo usuário. O algoritmo deve utilizar a fórmula de Baskara.

Solução:



Obs: Este algoritmo poderia ser implementado para prever o cálculo das raízes complexas da equação.

Os fluxogramas são elaborados tendo como base três tipos de estruturas de programação. São elas: estruturas seqüenciais, estruturas de decisão e estruturas de repetição. Nos exemplos até aqui apresentados foram utilizados apenas dois tipos destas estruturas: seqüenciais e de decisão.

- **Estruturas seqüenciais:** este tipo de estrutura é representado pela conexão de dois ou mais símbolos de processamento conectados por uma linha (ou seta), ou seja, indicando o sentido do fluxo de execução. A Figura 2.2 ilustra um exemplo deste tipo de estrutura.

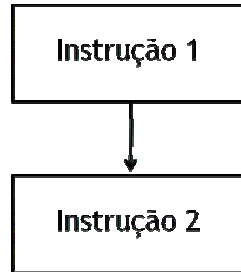


Figura 2.2: Representação de uma estrutura seqüencial.

- **Estruturas de seleção:** este tipo de estrutura é representado pelo uso combinado dos símbolos de decisão, de processamento e de indicação do sentido do fluxo de execução. A Figura 2.3 ilustra um exemplo deste tipo de estrutura.

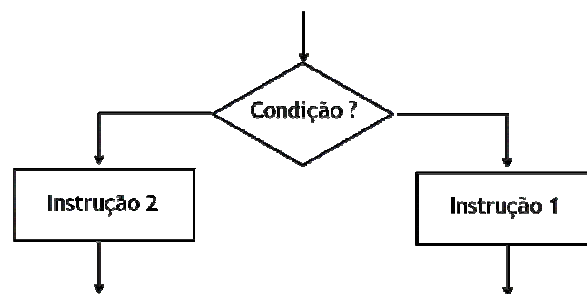


Figura 2.3: Representação de uma estrutura de seleção.

- **Estruturas de iteração:** este tipo de estrutura também é representado pelo uso combinado dos símbolos de decisão, de processamento e de indicação do sentido do fluxo de execução. A Figura 2.4 ilustra um exemplo deste tipo de estrutura.

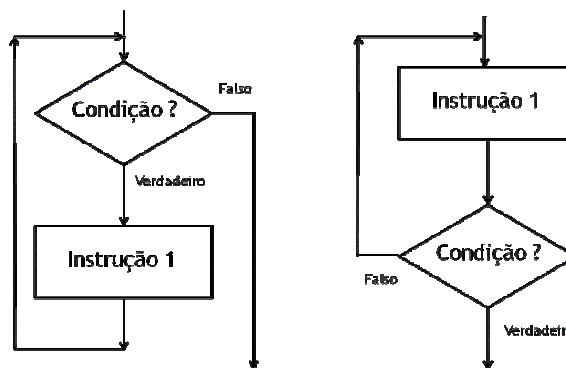


Figura 2.4: Representação de uma estrutura de iteração.



2.1.4. Exemplos de Pseudo-código

Neste item apresentaremos os exemplos já implementados na forma de pseudo-código.

Exemplo 4: Implementar na forma de pseudo-código algoritmo para calcular a área de um triângulo (area) empregando a fórmula de Herão (ou de Heron). São fornecidos os comprimentos dos três lados do triângulo (a, b, c).

Fórmula para cálculo da área: $area = \sqrt{s \cdot (s-a) \cdot (s-b) \cdot (s-c)}$

onde s é o semi-perímetro do triângulo, calculado por: $s = \frac{(a+b+c)}{2}$

* Observe que a ordem de cálculo é muito importante para resolução deste problema. Primeiro é preciso calcular o perímetro (s) e em seguida a área (area).

Solução:

Pseudo-código do algoritmo
Início
Ler (a, b, c)
$s \leftarrow (a+b+c)/3$
$area \leftarrow \sqrt{s \cdot (s-a) \cdot (s-b) \cdot (s-c)}$
Exibir (area)
Fim

Exemplo 5: Implementar na forma de pseudo-código algoritmo para o cálculo da média final (MF) dos alunos em uma disciplina do curso de Engenharia Ambiental. Nesta disciplina os alunos realizaram duas provas (P1 e P2) e um trabalho (T), e o seguinte critério de avaliação foi estabelecido pelo professor:

P1 - nota da primeira avaliação

P2 - nota da segunda avaliação

T - nota do trabalho realizado em equipe

MF - média final

Se $P1 \geq 5,0$ e $P2 \geq 5,0$ a média final é calculada por: $MF = \frac{(2 \cdot P1 + 2 \cdot P2 + 2 \cdot T)}{6}$

Caso contrário: $MF = \frac{(2 \cdot P1 + 2 \cdot P2 + T)}{5}$

Solução:

Pseudo-código do algoritmo
Início Ler (P1, P2, T) Se $P1 \geq 5,0$ e $P2 \geq 5,0$ Então $MF \leftarrow (2 \cdot P1 + 2 \cdot P2 + 2 \cdot T) / 6$ Senão $MF \leftarrow (2 \cdot P1 + 2 \cdot P2 + T) / 5$ Fim Se Exibir (MF) Fim

Exemplo 6: Formule algoritmo para calcular as raízes (x_1 e x_2) de uma equação do segundo grau ($a \cdot x^2 + b \cdot x + c = 0$) a partir dos valores das constantes a, b e c fornecidas pelo usuário. O algoritmo deve utilizar a fórmula de Báskara.

Solução:

Pseudo-código do algoritmo
Início Ler (a, b, c) Se $a = 0$ Então Exibir (“A equação não é do 2º grau”) Fim Se Fim $\Delta \leftarrow b^2 - 4 \cdot a \cdot c$ Se $\Delta < 0$ Então Exibir (“Não existem raízes reais”) Fim Se $\Delta = 0$ Então $x_1 \leftarrow \frac{-b}{2 \cdot a}$ $x_2 \leftarrow x_1$ Senão Se $\Delta > 0$ Então

$$x1 \leftarrow \frac{-b + \sqrt{\text{Delta}}}{2 \cdot a}$$

$$x2 \leftarrow \frac{-b - \sqrt{\text{Delta}}}{2 \cdot a}$$

Fim Se

Exibir (x1, x2)

Fim

Obs: A estrutura **Se** é finalizada com o termo **Fim Se** para evitar confusões.