

ASSOCIAÇÃO PARAIBANA DE ENSINO RENOVADO
FACULDADE PARAIBANA DE PROCESSAMENTO DE DADOS

TÉCNICAS DE PROGRAMAÇÃO



ALGORITMO

Capítulo 1 - ALGORITMOS

1.1. CONCEITO

A palavra **algoritmo**, à primeira vista, parece-nos estranha. Embora possua designação desconhecida, fazemos uso constantemente de algoritmos em nosso cotidiano: a maneira como uma pessoa toma banho é um algoritmo. Outros algoritmos freqüentemente encontrados são:

- instruções para se utilizar um aparelho eletrodoméstico;
- uma receita para preparo de algum prato;
- guia de preenchimento para declaração do imposto de renda;
- a regra para determinação de máximos e mínimos de funções por derivadas sucessivas;
- a maneira como as contas de água, luz e telefone são calculados mensalmente; etc.

São vários os conceitos para algoritmo. Escolhemos alguns para serem apresentados aqui:

“Um conjunto finito de regras que provê uma seqüência de operações para resolver um tipo de problema específico”
[KNUTH]

“Seqüência ordenada, e não ambígua, de passos que levam à solução de um dado problema”.
[TREMBLAY]

“Processo de cálculo, ou de resolução de um grupo de problemas semelhantes, em que se estipulam, com generalidade e sem restrições, as regras formais para a obtenção do resultado ou da solução do problema”.
[AURÉLIO]

1.2. POR QUE PRECISAMOS DE ALGORITMOS?

Vejam os que algumas pessoas importantes, para a Ciência da Computação, disseram a respeito de algoritmo:

“A noção de algoritmo é básica para toda a programação de computadores”.
[KNUTH - Professor da Universidade de Stanford, autor da coleção “The art of computer programming”]

“O conceito central da programação e da ciência da computação é o conceito de algoritmo”.
[WIRTH - Professor da Universidade de Zurique, autor de diversos livros na área e responsável pela criação de linguagens de programação como ALGOL, PASCAL e MODULA-2]

A importância do algoritmo está no fato de termos que especificar uma seqüência de passos lógicos para que o computador possa executar uma tarefa qualquer, pois o mesmo por si só não tem vontade própria, faz apenas o que mandamos. Com uma ferramenta algorítmica, podemos conceber uma solução para um dado problema, independentemente de uma linguagem específica e até mesmo do próprio computador.

1.3. CARACTERÍSTICAS

Todo algoritmo deve apresentar algumas características básicas:

- ter fim;
- não dar margem à dupla interpretação (não ambíguo);
- capacidade de receber dado(s) de entrada do mundo exterior;
- poder gerar informações de saída para o mundo externo ao do ambiente do algoritmo;
- ser efetivo (todas as etapas especificadas no algoritmo devem ser alcançáveis em um tempo finito).

1.4. FORMAS DE REPRESENTAÇÃO

Algoritmos podem ser representados, dentre outras maneiras, por:

1.4.1. DESCRIÇÃO NARRATIVA

Faz-se uso do português para descrever algoritmos.

EXEMPLO: Receita de Bolo:

Providencie manteiga, ovos, 2 Kg de massa, etc.
Misture os ingredientes
Despeje a mistura na fôrma de bolo
Leve a fôrma ao forno
Espere 20 minutos
Retire a fôrma do forno
Deixe esfriar
Prove

VANTAGENS:

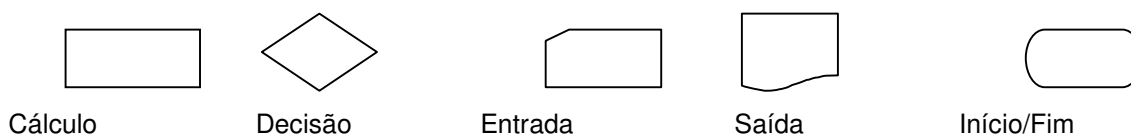
- o português é bastante conhecido por nós;

DESVANTAGENS:

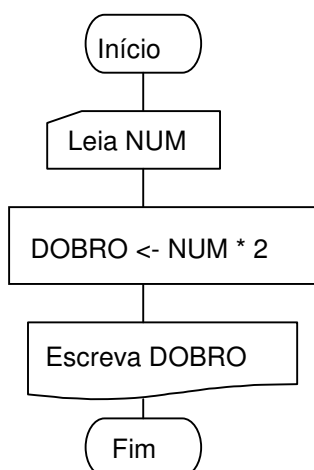
- imprecisão;
- pouca confiabilidade (a imprecisão acarreta a desconfiança);
- extensão (normalmente, escreve-se muito para dizer pouca coisa).

1.4.2. FLUXOGRAMA

Utilização de símbolos gráficos para representar algoritmos. No fluxograma existem símbolos padronizados para início, entrada de dados, cálculos, saída de dados, fim, etc.



EXEMPLO



EXPLICAÇÃO

Início do algoritmo

Entrada do número

Cálculo do dobro do número

Apresentação do resultado

Fim do algoritmo

VANTAGENS:

- Uma das ferramentas mais conhecidas;
- Figuras dizem muito mais que palavras;
- Padrão mundial

DESVANTAGENS:

- Faz com que a solução do problema já esteja amarrada a dispositivos físicos;

- Pouca atenção aos dados, não oferecendo recursos para descrevê-los ou representá-los;
- Complica-se à medida que o algoritmo cresce.

1.4.3. LINGUAGEM ALGORÍTMICA

Consiste na definição de uma pseudolinguagem de programação, cujos comandos são em português, para representar algoritmos.

EXEMPLO: Algoritmo CALCULA_DOBRO
 início
 Leia NUM
 DOBRO \leftarrow 2 * NUM
 Escreva DOBRO
 fim

VANTAGENS:

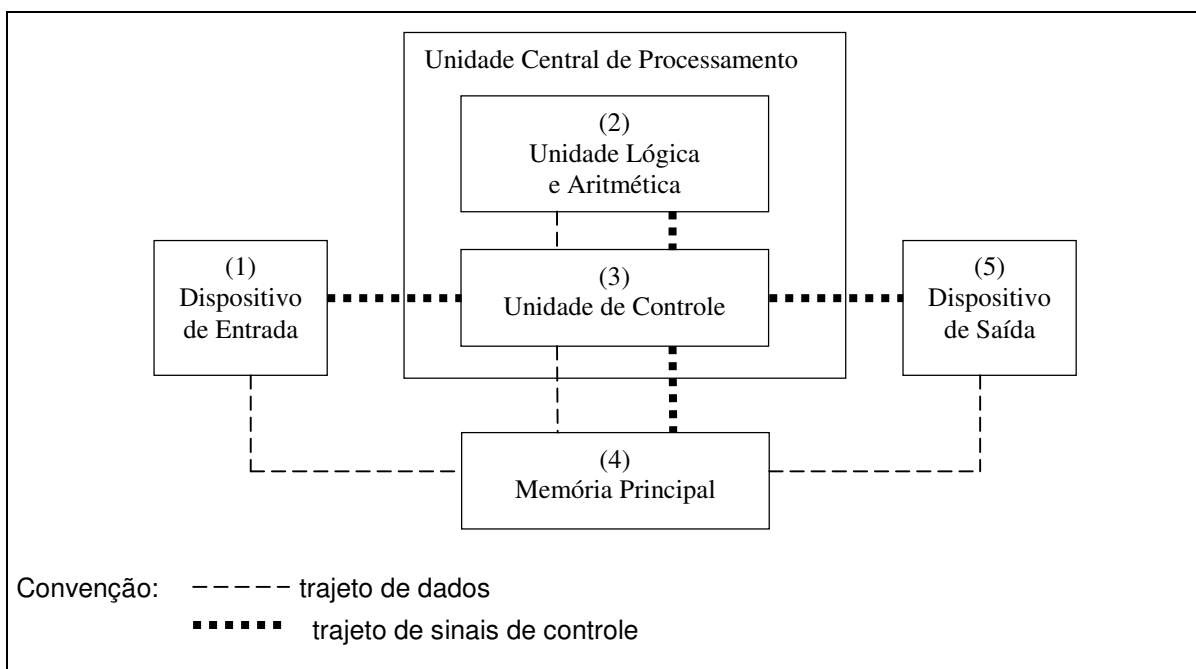
- Independência física da solução (solução lógica apenas);
- Usa o português como base;
- Pode-se definir quais e como os dados vão estar estruturados;
- Passagem quase imediata do algoritmo para uma linguagem de programação qualquer.

DESVANTAGENS:

- Exige a definição de uma linguagem não real para trabalho;
- Não padronizado.

1.5. UM AMBIENTE PARA ESCREVER ALGORITMOS

Descreveremos uma máquina hipotética para a qual escreveremos nossos algoritmos. O nosso computador hipotético apresentará a seguinte organização:



Cada uma das partes constituintes da figura acima tem os seguintes significados:

- (1) Dispositivo de entrada (o teclado):
 É o meio pelo qual os dados que serão trabalhados pelo algoritmo vão ser introduzidos em nosso computador hipotético;
- (2) Unidade Lógica e Aritmética (ULA):
 Parte responsável pelas operações matemáticas e avaliações lógicas;

- (3) Unidade de Controle:
Exerce controle sobre as demais partes do nosso computador. É uma verdadeira gerente que distribui tarefas às outras unidades;
- (4) Memória:
Guarda o algoritmo a ser executado e os dados a serem utilizados pelo mesmo. Todo dado fornecido ao computador e o resultado de suas operações ficam guardados na memória;
- (5) Dispositivo de Saída (vídeo e impressora):
É o meio que se dispõe para apresentação dos resultados obtidos.

1.5.1. FUNCIONAMENTO DO NOSSO COMPUTADOR

Todos os computadores, independentemente dos seus tamanhos, são conceitualmente semelhantes ao esquema da figura anterior (há algumas diferenças, mas não trataremos aqui dos casos especiais).

Resumidamente, podemos afirmar que existem 4 (quatro) operações básicas que qualquer computador pode executar:

- a) **operações de entrada e saída:** ler dados do teclado e escrever dados na tela são exemplos destas operações. Elas servem para introduzir dados na memória do nosso computador e exibir dados que já estejam lá armazenados;
- b) **operações aritméticas:** são utilizadas na realização de operações matemáticas (adição, subtração, multiplicação e divisão);
- c) **operações lógicas e relacionais:** têm aplicabilidade em comparações, testes de condições lógicas ($2 > 6$? $X = Y$?);
- d) **movimentação de dados entre os vários componentes:** as operações aritméticas são executadas na Unidade Lógica e Aritmética, necessitando da transferência dos dados para essa unidade e da volta do resultado final para ser guardado na memória.

1.5.2. RESOLVENDO UM PROBLEMA

Suponha que queiramos resolver o seguinte problema: a partir de dois números que serão informados, calcular a adição dos mesmos. Se você fosse encarregado de efetuar essa tarefa, seria bem provável que utilizasse os passos a seguir:

- a) saber quais são os números;
- b) calcular a soma dos números;
- c) responder à questão com o valor do resultado.

Vejamos como seria resolvido esse mesmo problema em termos das operações básicas citadas anteriormente:

- a) operação de entrada de dados dos números ;
- b1) movimento do valor dos números entre a memória e a ULA;
- b2) operação aritmética de somar os 2 números;
- b3) movimentação do resultado da ULA para guardar na memória;
- c) operação de saída do resultado, que está guardado na memória, para o dispositivo de saída desejado.

Deve-se salientar que os passos b1 e b3, normalmente, ficam embutidos na operação matemática, não sendo explicitados.

Em resumo, pode-se dizer que escrever algoritmos ou, em última análise, programar, consiste em dividir qualquer problema em muitos pequenos **passos**, usando uma ou mais das quatro operações básicas citadas.

Esses passos que compõem o algoritmo são denominados de **comandos**. Os comandos de uma linguagem de programação podem estar mais próximos da máquina (linguagens de baixo nível) ou serem mais facilmente entendidos pelo homem (linguagens de alto nível). A seqüência de operações básicas, dada anteriormente, para resolver o problema de adicionar dois números, está em uma linguagem de baixo nível para o nosso computador hipotético. Em uma linguagem de alto nível teríamos um resultado assim:

```
Leia X,Y  
SOMA ← X + Y  
Escreva SOMA
```

1.6. ESTRUTURAS CHAVES DA CONSTRUÇÃO DE ALGORITMOS

Existem 3 estruturas básicas de controle nas quais se baseiam os algoritmos: sequenciação, decisão e repetição. Detalharemos cada uma delas:

1.6.1. SEQUENCIAÇÃO

Os comandos do algoritmo fazem parte de uma seqüência, onde é relevante a ordem na qual se encontram os mesmos, pois serão executados um de cada vez, estritamente, de acordo com essa ordem. De uma forma genérica, poderíamos expressar uma seqüência da seguinte maneira:

```
Comando-1  
Comando-2  
Comando-3  
:  
Comando-n
```

Tem-se uma sequenciação de **n** comandos na qual os comandos serão executados na ordem em que aparecem, isto é, o comando de ordem $i+1$ só será executado após a execução do de ordem i (o 3º só será executado após o 2º).

Todo algoritmo é uma seqüência. A sequenciação é aplicada quando a solução do problema pode ser decomposta em passos individuais.

1.6.2. DECISÃO OU SELEÇÃO

Essa estrutura também é conhecida por estrutura condicional. Há a subordinação da execução de um ou mais comandos à veracidade de uma condição. Vejamos o funcionamento:

```
Se <condição> Então  
    <comando-1>  
Senão  
    <comando-2>
```

Se a <condição> for verdadeira será executado o <comando-1> e, em caso contrário, teremos a execução de <comando-2>.

A decisão deve ser sempre usada quando há a necessidade de testar alguma condição e em função da mesma tomar uma atitude. Em nosso dia-a-dia, estamos sempre tomando decisões, vejamos um exemplo:

```
Se tiver dinheiro suficiente, então vou almoçar em um bom restaurante.  
Caso contrário (senão), vou comer um sanduíche na lanchonete da esquina.
```

1.6.3. REPETIÇÃO OU ITERAÇÃO

Essa estrutura também é conhecida por “looping” ou laço. A repetição permite que tarefas individuais sejam repetidas um número determinado de vezes ou tantas vezes quantas uma condição lógica permita. Vejamos alguns exemplos:

- a) vou atirar pedras na vidraça até quebrá-la;
- b) baterei cinco pênaltis;
- c) enquanto tiver saúde e dinheiro, vou desfrutar a vida.

No exemplo (a), vai-se repetir a ação de atirar pedras na janela até que seja satisfeita a condição de quebrar a janela.

No exemplo (b), haverá a repetição da atitude de bater um pênalti um número determinado de vezes (cinco).

No exemplo (c), a condição que me permitirá continuar desfrutando a vida é ter dinheiro e saúde.

A utilização combinada dessas 3 estruturas descritas vai permitir expressar, usando qualquer que seja a ferramenta, a solução para uma gama muito grande de problemas. Todas as linguagens de programação oferecem representantes dessas estruturas.

1.7. REFINAMENTOS SUCESSIVOS

Um algoritmo é considerado completo se os seus comandos forem do entendimento do seu destinatário.

Num algoritmos, um comando que não for do entendimento do destinatário terá que ser desdobrado em novos comandos, que constituirão um **refinamento** do comando inicial, e assim sucessivamente, até que os comandos sejam entendidos pelo destinatário.

Por exemplo, o algoritmo para calcular a média aritmética de dois números pode ser escrito da seguinte forma:

```
Algoritmo CALCULA_MÉDIA
Início
  Receba os dois números
  Calcule a média dos dois números
  Exiba o resultado
Fim
```

Podemos desdobrar o comando “Calcule a média dos dois números” em:

```
Soma os dois números
Divida o resultado por 2
```

Após esse refinamento, o algoritmo pode ser considerado completo, a menos que o destinatário não saiba fazer as operações de adição e divisão, ou não seja capaz de entender diretamente algum comando.

O algoritmo estando completo, podemos reescrevê-lo, inserindo o refinamento na posição do comando que foi refinado. Assim sendo, obtém-se:

```
Algoritmo CALCULA_MÉDIA
Início
  Receba os dois números
  Soma os dois números
  Divida o resultado por 2
  Exiba o resultado
Fim
```

Reescrever um algoritmo completo, com os refinamentos sucessivos inseridos nos seus devidos lugares, permite ter uma visão global de como o algoritmo deve ser executado.

À medida que o algoritmo passa a ser maior e mais complexo, esta visão global torna-se menos clara e, neste caso, um algoritmo apresentado com os refinamentos sucessivos separados oferece uma melhor abordagem para quem precisar entendê-lo.

EXERCÍCIOS PROPOSTOS

- P1.01. Defina, com suas palavras, o que é algoritmo.
- P1.02. Cite alguns algoritmos que podemos encontrar na vida cotidiana.
- P1.03. De acordo com seu entendimento, qual é a característica mais importante em um algoritmo? Justifique a sua resposta.
- P1.04. Um algoritmo não pode conter um comando como "Escreva todos os números inteiros positivos". Por quê?
- P1.05. Suponha que temos um robô a nossa disposição. Esse robô chama-se MANNY e precisa ser ensinado a fazer determinadas tarefas. Para ensinar o MANNY, vamos fazer uso do português para passar-lhe as instruções necessárias à execução de cada atividade. Escreva os passos necessários para o nosso robô executar:
- a) encher uma bacia com água;
 - b) trocar uma lâmpada no teto de sua casa;
 - c) trocar o pneu de um carro;
 - d) calcular a sua idade daqui a 20 anos;
 - e) calcular a média de um aluno com 3 notas.
- P1.06. Cite as formas básicas para se representar algoritmos, definindo-as.
- P1.07. Em sua opinião, qual a melhor forma de se representar algoritmos? Justifique sua resposta.
- P1.08. Descreva, com suas próprias palavras, o funcionamento do nosso computador hipotético.
- P1.09. Especifique soluções, em termos das operações básicas do nosso computador, para os itens (d) e (e) do exercício P1.05.
- P1.10. Quais as estruturas básicas de controle dos algoritmos? Explique cada uma delas.
- P1.11. Identifique nas respostas do exercício P1.05 a utilização das estruturas básicas de controle de fluxo.
- P1.12. Escreva o algoritmo solução para o problema de multiplicar dois números (a solução deve ser expressa em alto nível).
- P1.13. Resolva o P1.09 em termos de uma linguagem de alto nível.
- P1.14. Em que consiste a técnica de "refinamentos sucessivos" ?
- P1.15. É comum ouvirmos programadores experientados afirmarem:
"algoritmos ... aprendi e nunca usei na prática ... não vejo necessidade...".
Discuta esse tipo de afirmativa.

Capítulo 2

LINGUAGEM ALGORÍTMICA

Para uma melhor padronização de nossos estudos, vamos agora definir uma linguagem para a construção de nossos algoritmos. Antes de começarmos a definir as operações básicas de nossa linguagem algorítmica, é importante que conheçamos o conceito de *variável*.

2.1. CONCEITO DE VARIÁVEL

Sabe-se da Matemática que uma variável é a representação simbólica dos elementos de um certo conjunto.

Nos algoritmos destinados a resolver um problema no computador, **a cada variável corresponde uma posição de memória, cujo conteúdo pode variar ao longo do tempo durante a execução de um algoritmo**. Embora a variável possa assumir diferentes valores, ela só pode armazenar um valor a cada instante.

Toda variável é identificada por um nome ou identificador. Assim, por exemplo, num algoritmo para calcular a área de um triângulo retângulo pelo teorema de pitágoras ($a^2 = b^2 + c^2$), os identificadores A, B e C podem representar as posições de memória que armazenam o valor da hipotenusa e dos catetos. É importante que nunca usemos uma palavra reservada, isto é, que faça parte da linguagem algorítmica, como um identificador, pois poderá causar ambigüidade no entendimento do algoritmo.

Na nossa linguagem algorítmica, vamos admitir que nossas variáveis poderão armazenar valores numéricos ou alfanuméricos (strings).

Agora que sabemos o que é uma variável, vamos às definições das operações básicas de nossa linguagem algorítmica.

2.2. OPERAÇÃO DE ATRIBUIÇÃO

A operação de atribuição permite que se forneça um valor a uma certa variável. Se for atribuído uma expressão à variável, será armazenado o resultado daquela expressão. Se for atribuído uma outra variável, será armazenado o conteúdo daquela variável. Para a operação de atribuição, utilizaremos a seguinte sintaxe:

variável \leftarrow expressão

Exemplos:

A \leftarrow 2

B \leftarrow A

NOTA \leftarrow 10

NOME \leftarrow 'João'

SENHA \leftarrow 'X3Y9'

C \leftarrow 1 / 3

A \leftarrow B + C

NOTA \leftarrow NOTA - 1

X \leftarrow 2.5

2.3. OPERAÇÕES DE ENTRADA E SAÍDA

Os cálculos do computador são de pouco valor a não ser que, primeiro, possamos fornecer os dados sobre os quais estes cálculos serão efetuados e, segundo, ver os resultados destes cálculos.

Definimos aqui dois novos comandos da nossa linguagem algorítmica para manusear entrada e saída. O comando **leia** nos permite ler valores dados atribuindo-os à variáveis indicadas; o comando **escreva** nos permite mostrar os resultados. A entrada pode vir do teclado ou de qualquer outro dispositivo de entrada. A saída pode aparecer na tela do monitor de vídeo ou ser impressa em papel. Não nos preocuparemos com detalhes destes dispositivos. A sintaxe destes comandos são:

leia variável-1, variável-2, ..., variável-n

escreva expressão-1, expressão-2, ..., expressão-n

Exemplos:

leia NOTA

leia A,B,C

leia NOME

escreva 15

escreva NOTA+2

escreva A,B

EXERCÍCIOS PROPOSTOS

P2.01. Dê o conceito de variável.

P2.02. Exemplifique o uso da operação de Atribuição.

P2.03. Qual a finalidade de uma operação de entrada ? Dê exemplos.

P2.04. Qual a finalidade de uma operação de saída? Dê exemplos.

P2.05. Escreva os comandos necessários para:

- a) ler o nome de uma pessoa
- b) ler as 3 notas de um aluno
- c) ler o peso e altura de uma pessoa

2.4. ESTRUTURA SEQÜENCIAL

Num algoritmo, os comandos deverão ser executados numa seqüência linear, seguindo-se o texto em que estão escritos, de cima para baixo, se não houver indicação em contrário.

Nesta apostila, os algoritmos são iniciados com a palavra **início** e finalizados com a palavra **fim**.

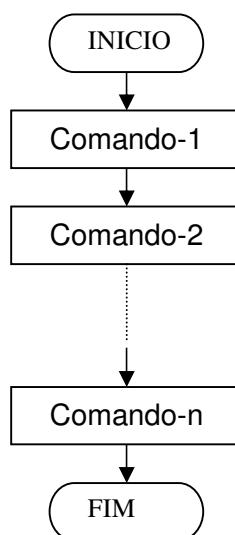
Linguagem Algorítmica:

```
início  
  comando-1  
  comando-2  
  ...  
  comando-n  
fim
```

Exemplo:

```
início  
  leia A,B  
  SOMA  $\leftarrow$  A + B  
  escreva SOMA  
fim
```

Fluxograma:



EXERCÍCIOS PROPOSTOS

P2.06. Em que consiste a estrutura seqüencial?

P2.07. Determine os valores finais de A, B e C após a execução do trecho do algoritmo abaixo:

<pre>A \leftarrow 0 B \leftarrow 1 C \leftarrow A + B A \leftarrow A + 1 B \leftarrow A + B + C</pre>
--

A	B	C

P2.08. A ordem das atribuições é importante? $A \leftarrow B$ e $C \leftarrow A$ tem o mesmo efeito de $C \leftarrow A$ e $A \leftarrow B$?

P2.09. Em quais dos seguintes pares é importante a ordem dos comandos ?

- | | | | |
|---|---|---|---|
| a) $X \leftarrow Y$
$Y \leftarrow X$ | b) $X \leftarrow Y$
$Z \leftarrow X$ | c) $X \leftarrow Z$
$X \leftarrow Y$ | d) $Z \leftarrow Y$
$X \leftarrow Y$ |
|---|---|---|---|

P2.10. Escreva um algoritmo que leia duas variáveis inteiras e troque o conteúdo entre elas.

P2.11. Escreva um algoritmo que leia um número inteiro positivo e exiba o dobro do mesmo.

P2.12. Escreva um algoritmo para calcular e exibir a média ponderada de 2 notas dadas.
(nota1= peso 6 e nota2= peso 4)

P2.13. Escreva um algoritmo para calcular e exibir o comprimento de uma circunferência, sendo dada o valor de seu raio.

$$C = 2\pi R$$

P2.14. Escreva um algoritmo para ler uma temperatura dada na escala Fahrenheit e exibir o equivalente em Celsius.

$$C = \frac{5}{9}(F - 32)$$

P2.15. Escreva um algoritmo para calcular a área de um triângulo, sendo dados a sua base e a sua altura.

$$ÁREA = \frac{BASE \cdot ALTURA}{2}$$

2.5. ESTRUTURA CONDICIONAL

A estrutura condicional permite a escolha do grupo de ações e estruturas a ser executado quando determinadas condições (expressões lógicas) são ou não satisfeitas.

Esta estrutura pode se apresentar de duas formas.

2.5.1. ESTRUTURA CONDICIONAL SIMPLES

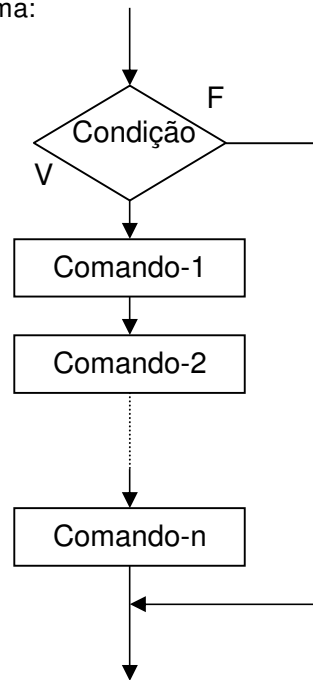
Linguagem Algorítmica:

```
se condição então  
  comando-1  
  comando-2  
  ...  
  comando-n
```

Exemplo:

```
...  
se MÉDIA  $\geq$  7 então  
  SITUAÇÃO  $\leftarrow$  'Aprovado'  
  Escreva SITUAÇÃO  
...
```

Fluxograma:



2.5.2. ESTRUTURA CONDICIONAL COMPOSTA

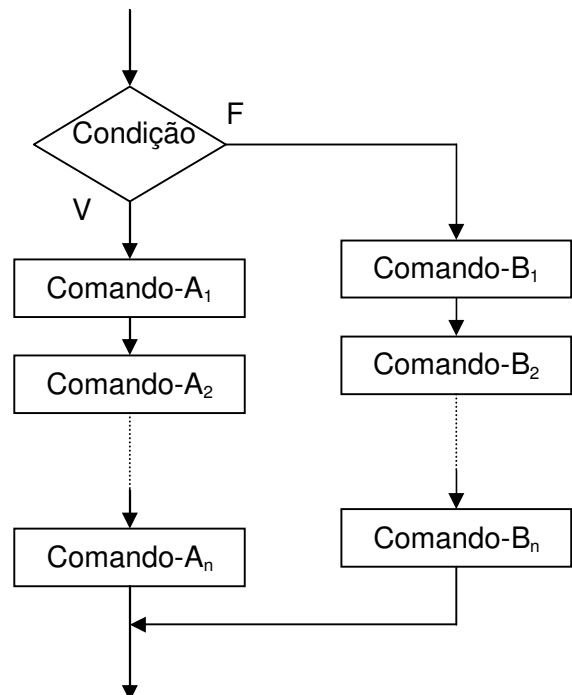
Linguagem Algorítmica:

```
se condição então  
  comando-A1  
  comando-A2  
  ...  
  comando-An  
senão  
  comando-B1  
  comando-B2  
  ...  
  comando-Bn
```

Exemplo:

```
...  
se PESO > 80 então  
  escreva 'Você está obeso'  
  escreva 'Faça atividades físicas'  
senão  
  escreva 'Você está no peso certo'  
  escreva 'Procure manter sua forma'  
...
```

Fluxograma:



EXERCÍCIOS PROPOSTOS

P2.16. Qual a utilidade da estrutura condicional ?

P2.17. Qual a diferença entre a estrutura condicional simples e a composta ?

P2.18. Escreva um algoritmo para ler um número e determinar se ele é maior, igual ou menor que zero.

P2.19. Escreva um algoritmo que leia dois números e exiba-os em ordem crescente.

P2.20. Escreva um algoritmo que leia dois números e exiba o maior deles.

P2.21. Deseja-se calcular a conta de consumo de energia elétrica de um consumidor. Para isto, escreva um algoritmo que leia o código do consumidor, o preço do Kw e a quantidade de Kw consumido, e exiba o código do consumidor e o total a pagar.

- total a pagar = preço x quantidade

- total a pagar mínimo = R\$ 11,20

P2.22. Escreva um algoritmo que determine o grau de obesidade de uma pessoa, sendo fornecido o peso e a altura da pessoa. O grau de obesidade é determinado pelo índice da massa corpórea ($\text{Massa} = \text{Peso} / \text{Altura}^2$) através da tabela abaixo:

MASSA CORPÓ- REA	GRAU DE OBESI- DADE
< 26	Normal
≥ 26 e < 30	Obeso
≥ 30	Obeso Mórbido

P2.23. Faça um algoritmo que, dado as três notas de um aluno, determine e exiba a sua média final e o seu conceito, sabendo-se que:

- a média final é calculada pela média aritmética das 3 notas;

- o conceito é determinado de com base na tabela abaixo:

MÉDIA FINAL	CONCEITO
$\geq 8,0$	A
$\geq 5,0$ e < 8,0	B
< 5,0	C

P2.24. O Botafogo Futebol Clube deseja aumentar o salário de seus jogadores. O reajuste deve obedecer a seguinte tabela:

SALÁRIO ATUAL (R\$)	AUMENTO
0,00 a 1.000,00	20%
1.000,01 a 5.000,00	10%
acima de 5.000,00	0%

Escrever um algoritmo que leia o nome e o salário atual de um jogador, e exiba o nome, o salário atual e o salário reajustado.

P2.25. Faça um algoritmo para calcular a conta final de um hóspede de um hotel fictício, considerando que:

a) serão lidos o nome do hóspede, o tipo do apartamento utilizado (A, B, C ou D), o número de diárias utilizadas pelo hóspede e o valor do consumo interno do hóspede;

b) o valor da diária é determinado pela seguinte tabela:

TIPO DO APTO.	VALOR DA DIÁRIA (R\$)
A	150,00
B	100,00
C	75,00
D	50,00

- c) o valor total das diárias é calculado pela multiplicação do número de diárias utilizadas pelo valor da diária;
- d) o subtotal é calculado pela soma do valor total das diárias e o valor do consumo interno;
- e) o valor da taxa de serviço equivale a 10% do subtotal;
- f) a total geral resulta da soma do subtotal com a taxa de serviço.
- g) escreva a conta final contendo: o nome do hóspede, o tipo do apartamento, o número de diárias utilizadas, o valor unitário da diária, o valor total das diárias, o valor do consumo interno, o subtotal, o valor da taxa de serviço e o total geral.

2.6. ESTRUTURA DE REPETIÇÃO

A estrutura de repetição permite que uma sequência de comandos seja executada repetidamente até que uma determinada condição não seja satisfeita. Utilizaremos o comando **enquanto** para representar esta estrutura. Sua sintaxe é:

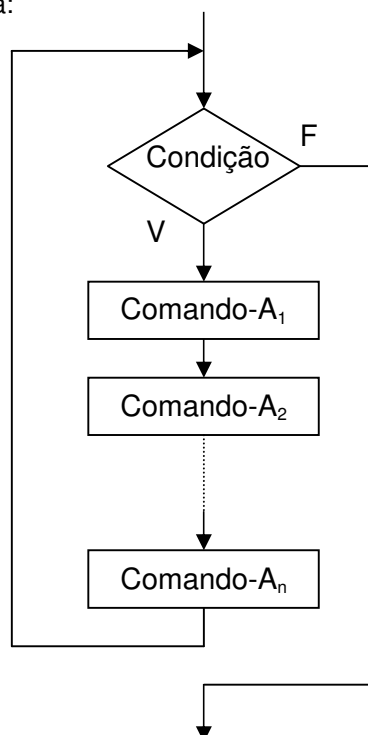
Linguagem Algorítmica:

```

enquanto condição
  comando-A1
  comando-A2
  ...
  comando-An

```

Fluxograma:



Exemplo:

```

...
leia SENHA
enquanto SENHA <> 'ASPER'
  escreva 'Senha inválida'
  escreva 'Digite a senha novamente'
  leia SENHA
...

```

EXERCÍCIOS PROPOSTOS

P2.26. Qual a utilidade da estrutura de repetição?

P2.27. Faça o acompanhamento da execução do trecho de algoritmo abaixo e preencha a Tabela de Variáveis:

TRECHO DE ALGORITMO
$N \leftarrow 0$ $L \leftarrow 1$ enquanto $N \neq 6$ $L \leftarrow L \cdot (-1)$ $N \leftarrow N + 1$ se $L > 0$ então escreva N

TABELA DE VARIÁVEIS			
N	L	$N \neq 6$	Saída

- P2.28. Faça um algoritmo que mostre todos os números inteiros de 1 a 50.
- P2.29. Faça um algoritmo que leia um número N, some todos os números inteiros de 1 a N, e mostre o resultado obtido.
- P2.30. Faça um algoritmo que leia uma lista de números inteiros positivos terminada pelo número 0 (zero). Ao final, o algoritmo deve mostrar a média aritmética de todos os números lidos (excluindo o zero).
- P2.31. Faça um algoritmo que leia N e uma lista de N números e mostre a soma de todos os números da lista.
- P2.32. Escreva um algoritmo que leia um conjunto de 100 números inteiros positivos e determine o maior deles.
- P2.33. Escreva um algoritmo que leia um número N e uma lista de N números inteiros positivos e determine o maior número da lista.
- P2.34. Escreva um algoritmo que leia um conjunto de números inteiros positivos e determine o maior deles. A leitura do valor 0 (zero) indica o fim dos dados (flag).
- P2.35. Faça um algoritmo que gere a seguinte série: 10, 20, 30, 40, ..., 990, 1000.

$$H = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \dots + \frac{1}{N}$$

- P2.36. Sendo $H = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \dots + \frac{1}{N}$, faça um algoritmo p/ calcular H. O número N é lido.
- P2.37. Faça um algoritmo que leia um número N, calcule e mostre os N primeiros termos da sequência de Fibonacci (0, 1, 1, 2, 3, 5, 8, 13, ...). O valor lido para N sempre será maior ou igual a 2.
- P2.38. Escreva um algoritmo que calcule o fatorial de um número inteiro lido, sabendo-se que:
- $$N! = 1 \times 2 \times 3 \times \dots \times N-1 \times N$$
- $$0! = 1$$
- P2.39. O Botafogo Futebol Clube deseja aumentar o salário de seus 22 jogadores. O reajuste deve obedecer a seguinte tabela:

SALÁRIO ATUAL (R\$)	AUMENTO
0,00 a 1.000,00	20%
1.000,01 a 5.000,00	10%
acima de 5.000,00	0%

Escrever um algoritmo que:

- leia o nome e o salário atual de cada jogador;
- exiba o nome, o salário atual e o salário reajustado de cada jogador;
- exiba o total da folha de salários do clube, antes do reajuste.
- exiba o total da folha de salários do clube, após o reajuste.
- exiba o percentual de reajuste sobre o total da folha de salários.

- P2.40. O cardápio de uma casa de lanches, especializada em sanduíches, é dado abaixo. Escreva um algoritmo que leia o código e a quantidade de cada item comprado por um freguês, calcule e exiba o total a pagar. Obs: A leitura do código 'X' indica o fim dos itens.

CÓDIGO	PRODUTO	PREÇO (R\$)
H	Hamburger	1,50
C	Cheeseburger	1,80
M	Misto Quente	1,20
A	Americano	2,00
Q	Queijo Prato	1,00

- P2.41. Num frigorífico existem 90 bois. Cada boi traz preso em seu pescoço um cartão contendo seu número de identificação e seu peso. Faça um algoritmo que escreva o número e o peso do boi mais gordo e do boi mais magro (supondo que não haja empates).

- P2.42. Faça um algoritmo que leia a altura de um grupo de 20 pessoas, calcule e exiba:
- a maior altura do grupo;
 - a altura média;
 - o número de pessoas com altura superior a 2 metros.
- P2.43. Faça um algoritmo que, para um número indeterminado de pessoas:
- leia a idade de cada pessoa, sendo que a leitura da idade 0 (zero) indica o fim dos dados (flag) e não deve ser considerada;
 - calcule e escreva o número de pessoas;
 - calcule e escreva a idade média do grupo;
 - calcule e escreva a menor idade e a maior idade.
- P2.44. Faça um algoritmo que leia uma lista de letras terminada pela letra Z. Ao final, o algoritmo deve mostrar a quantidade lida de cada vogal.
- P2.45. Uma certa firma fez uma pesquisa de mercado para saber se as pessoas gostaram ou não de um novo produto lançado no mercado. Para isto, forneceu o sexo do entrevistado (M-masculino ou F-feminino) e sua resposta (S-sim ou N-não). Sabendo-se que foram entrevistadas 2.000 pessoas, fazer um algoritmo que calcule e escreva:
- número de pessoas que responderam sim (S);
 - número de pessoas que responderam não (N);
 - a porcentagem de pessoas do sexo feminino (F);
 - a porcentagem de pessoas do sexo masculino (M);
 - a porcentagem de pessoas do sexo feminino (F) que responderam sim (S);
 - a porcentagem de pessoas do sexo masculino (M) que responderam não (N).
- P2.46. Foi feita uma pesquisa de audiência de canal de TV em várias casas de uma certa cidade, num determinado dia. Para cada casa visitada, é fornecido o número do canal (5, 7, 10 ou 12) e o número de pessoas que o estavam assistindo naquela casa. Fazer um algoritmo que:
- leia um número indeterminado de dados, sendo que o flag corresponde ao número de canal igual a 0 (zero);
 - calcule e escreva a porcentagem de audiência de cada emissora.
- P2.47. Escreva um algoritmo que leia o número de andares de um prédio e, a seguir, para cada andar do prédio, leia o número de pessoas que entraram e saíram do elevador.
Considere que o elevador está vazio e está subindo, os dados se referem a apenas uma subida do elevador e que o número de pessoas dentro do elevador será sempre maior ou igual a zero.
Se o número de pessoas, após a entrada e saída, for maior que 15, deve ser mostrada a mensagem "Excesso de passageiros. Devem sair X", sendo X o número de pessoas que devem sair do elevador, de modo que seja obedecido o limite de 15 passageiros.
Após a entrada e saída no último andar, o algoritmo deve mostrar quantas pessoas permaneceram no elevador para descer.
- P2.48. Faça um algoritmo que leia vários códigos do jogador (1 ou 2) que ganhou o ponto em uma partida de pingue-pongue, e responda quem ganha a partida.
A partida chega ao final se um dos jogadores chega a 21 pontos e a diferença de pontos entre os jogadores é maior ou igual a dois. Caso contrário, ganha aquele que, com mais de 21 pontos, consiga colocar uma vantagem de dois pontos sobre o adversário.

