



Algoritmos 2023.2

Aula 2 – Variáveis, Constantes e Tipos de Dados

Profª Náthalee Cavalcanti de A. Lima
Email: nathalee.almeida@ufersa.edu.br
05/12/2023

Agenda



- Armazenamento de Dados
- Variáveis
- Constantes
- Tipos de Dados

Armazenamento de Dados



Na execução de um algoritmo ou de um programa, para receber dados de entrada, processar as instruções e fornecer os dados de saída, é necessário que o computador **ARMAZENE** esses dados e instruções em sua **MEMÓRIA**, a fim de poder **ACESSÁ-LOS** posteriormente.

Armazenamento de Dados



Exemplo:

Imagine que você e um amigo desejam calcular o algoritmo da média, ‘de cabeça’, sem fazer anotações.

Armazenamento de Dados



De forma similar, o computador precisa guardar os dados e instruções em endereços específicos de sua memória, durante a execução de um algoritmo, a fim de não ‘esquecer’ as informações, podendo, assim, processá-las eficazmente.



Variáveis

- Vimos que os dados utilizados são armazenados na memória do computador para serem acessados posteriormente.
- **Como acessar esses dados??**

Para acessar os dados, precisamos descobrir em que posição na memória do computador eles estão armazenados.

Variáveis



01 <u> </u> Maria	06 <u> </u> Francisco	11 <u> </u>
02 <u> </u> João	07 <u> </u>	12 <u> </u>
03 <u> </u> Sílvia	08 <u> </u> José	13 <u> </u>
04 <u> </u>	09 <u> </u>	14 <u> </u>
05 <u> </u>	10 <u> </u>	15 <u> </u>

Variável



Uma variável é uma posição na memória do computador, que é reservada para armazenar os dados que o algoritmo vai manipular.



Variável

- Uma variável precisa ter um nome (ou identificador) um tipo de dado associado a ela (tipo da variável) e a informação que ela armazena.
- O identificador serve para diferenciar a variável das demais, por isso deve ser único para cada variável.
- O processo de criação de uma variável é chamado de **declaração da variável**.

Variável



- Devem ser declaradas antes de serem utilizadas no algoritmo;
- Permite que seja reservado um espaço na memória para o dado que vai ser armazenado e utilizado.



Variáveis

- Como declarar uma variável?

<nome_da_variável> : <tipo_da_variável>

ou:

<lista_de_variáveis> : <tipo_da_variável>

Ex:

nota: real

idade: inteiro

sexo: caracter

nota1, nota2: real



Variáveis

- Quando declaramos as variáveis, elas estão “vazias”, isto é sem informações associadas a ela;
- Somente fornecemos informações à variável posteriormente no corpo do algoritmo;

Exemplo das gavetas:

- Para reservar uma gaveta para uma pessoa, não era necessário armazenar o conteúdo, de imediato, na gaveta.
- Só é possível armazenar conteúdo em uma gaveta que já foi previamente reservada. Ou seja, primeiro deve-se reservar uma gaveta e depois armazenar seu conteúdo.



Variável

- Uma variável pode assumir valores diferentes, mas só pode armazenar um valor a cada instante.
- O seu conteúdo pode mudar ao longo do tempo na execução do algoritmo.



Variáveis

Regras para definição dos nomes (identificadores) das variáveis.

1. Podem-se utilizar números e letras maiúsculas ou minúsculas.
2. Não se deve utilizar caracteres especiais, à exceção do caractere sublinhado, ou *underline* (_).
3. O primeiro caractere deve ser sempre letra ou sublinhado, mas nunca um número.
4. Não deve utilizar acentos gráficos, til ou cedilha. Além dos símbolos, não é permitido espaço em branco.
5. Não se devem utilizar as palavras reservadas da linguagem

Variáveis



1280

Arquivo Editar Exibir Algoritmo Código Ferramentas Ajuda

0.5 s

```
algoritmo "palavras reservadas"
var
inicio : literal
inicio
    inicio <- "Não utilize palavras reservadas para nomear variáveis!"
fimalgoritmo
```

Variáveis



Identificadores válidos:

- a) nome1
- b) ano_de_nascimento
- c) salario
- d) nota_aluno
- e) qtd_dias



Identificadores inválidos:

- a) lnome – não deve começar por número
- b) Ano de nascimento – não deve ter espaço em branco
- c) @salário\$ – não deve conter caracteres especiais (contém @ e \$)
- d) Nota*aluno/01 – não deve conter caracteres especiais (contém * e /)
- d) fim – não devemos utilizar palavras reservadas
- e) remédio – não deve utilizar acentos gráficos

Variáveis



Obs. 1: é importante que, ao identificar uma variável, você utilize palavras ou termos significativos. Por exemplo, para armazenar o valor do salário do mês janeiro, use `salario_janeiro`; para armazenar um número de telefone, use `num_fone`. Essa prática ajudará no entendimento do algoritmo (para quem lê), tornando-o autoexplicativo.

Obs. 2: apesar de não haver limitações para a quantidade de caracteres que comporão o nome de uma variável, não alongue desnecessariamente os nomes. Por exemplo: `salario_do_mes_de_janeiro_com_desconto`.

Obs. 3: ao utilizar nomes de variáveis com mais de uma palavra, separe as palavras com o caractere sublinhado. Por exemplo: `salario_do_mes`.

Obs. 4: ao declarar as variáveis de um algoritmo, primeiro analise o problema, depois defina todos os dados de entrada e saída e o tipo de cada um para, então, criar os identificadores. Isso o ajudará a se lembrar de declarar todas as variáveis necessárias à execução do algoritmo.



Constantes

- Tratam-se de valores fixos ou estáveis;
- Numéricos, caracteres e lógicos.

Exemplos – Numéricos

- a) 15
- b) 250
- c) 38.5

CONSTANTES



Exemplos - Caracteres ou literais

- a) “F”
- b) “Maria”
- c) “Avenida São José”

Exemplos – Lógicos

- a) Verdadeiro
- b) Falso

VARIÁVEIS E CONSTANTES



Ao atribuir valores às variáveis, podemos utilizar outras variáveis, expressões ou constantes. Veja:

algoritmo “exemplo”

var

a : inteiro

b : inteiro

c : inteiro

d : logico

f : literal

inicio

a <- 2 //atribuição de constante

b <- a + 1 //atribuição de expressão

c <- b //atribuição de variável

d <- falso //atribuição de constante

f <- “Teste” //atribuição de constante

fimalgoritmo

ATIVIDADE



1. Qual a importância da utilização das variáveis na criação de um algoritmo?
2. Por que é importante declarar uma variável antes de utilizá-la?
3. Por que o identificador de uma variável deve ser único?
4. Explique os problemas nos identificadores abaixo, sugerindo, em seguida, uma alternativa válida para cada um:
 %unidades
 anos sem aumento
 ah!
 Nota*do*aluno
 inteiro
 Numero_de_vezes_que_fui_ao_shopping_hoje
5. Explique com suas palavras o que é uma constante.

Dados Numéricos (Inteiros e Reais)



- **Inteiros**

Os números inteiros são os números positivos ou negativos que não possuem parte decimal ou fracionária. Ex.: 15, -487, 0, 27835, -14937, 100.

Dados Numéricos (Inteiros e Reais)



Exemplo:

ALGORITMO QUE CALCULA A IDADE DE UMA PESSOA, DADOS O ANO ATUAL E O ANO EM QUE ELA NASCEU.

algoritmo "calcula idade"

var

ano_atual : **inteiro**

ano : **inteiro**

idade_hoje : **inteiro**

inicio

 ano_atual <- 2023

leia (ano)

 idade_hoje <- ano_atual - ano

escreva (idade_hoje)

fimalgoritmo

Dados Numéricos (Inteiros e Reais)



- **Reais**

Os números reais são os números positivos ou negativos que englobam números decimais ou fracionários. Ex.: 15, -487, 1.78, 0.254, 27835, 100, 8.50.

São também chamados de pontos flutuantes, nas linguagens de programação.

Valores reais são aplicáveis em algoritmos que manipulam dados que expressam valores fracionários, como salário, média, preço, porcentagem, entre outros.

Dados Numéricos (Inteiros e Reais)



Exemplo

ALGORITMO QUE CALCULA A MÉDIA DO ALUNO

algoritmo "calcula média"

var

nota1 : **real**

nota2 : **real**

media : **real**

inicio

leia (nota1)

leia (nota2)

 media <- (nota1 + nota2)/2

escreva (media)

fimalgoritmo

Dados Numéricos (Inteiros e Reais)



Exemplo

ALGORITMO QUE CALCULA A MÉDIA DO ALUNO

algoritmo "calcula média"

var

nota1 : **real**

nota2 : **real**

media : **real**

inicio

leia (nota1)

leia (nota2)

 media <- (nota1 + nota2)/2

escreva (media)

fimalgoritmo

Dados Numéricos (Inteiros e Reais)



Obs.:

As casas decimais, nos números reais, devem ser separadas por ponto (.) e não por vírgula (,). Por exemplo: use “8.5” e não “8,5”.

Dados Literais



- Os dados literais são formados por um único caractere ou uma sequência de caracteres, que podem ser letras (maiúsculas ou minúsculas), números ou símbolos especiais (como #, \$, @, ?, &, entre outros).
- Os números, quando representados como caracteres, não podem ser utilizados para cálculos.
- Uma sequência de caracteres pode ser chamada também de cadeia de caracteres ou *string*.



Dados Literais

Exemplos: “Fone: 3222-2222”, “Av. Senador Salgado Filho, Nº 1550”, “João Silveira”, ‘M’, “152”, “CEP: 59052-250”, ‘F’.

algoritmo "literais"

var

nome : **literal**

sobrenome : **literal**

inicio

nome <- "Maria"

escreva ("Digite o sobrenome:")

leia (sobrenome)

escreva (nome, " ",sobrenome)

fimalgoritmo

Dados Literais



Veja outro exemplo, com um caractere isolado:

algoritmo "caractere"

var

sexo : **caractere**

inicio

 sexo <- 'F'

escreva (sexo)

fimalgoritmo

Para um caractere isolado, utilizaremos a notação *character* ou *caractere*.



Dados Lógicos

- Os dados lógicos são também chamados de *booleanos*, por ter sua origem na álgebra *booleana*.
- Os valores que esses tipos de dados podem assumir são *Verdadeiro* ou *Falso*, podendo representar apenas um desses dois valores.

Dados Lógicos



algoritmo "logicos"

var

igualdade : logico

inicio

igualdade <- 12 = 10

escreva (igualdade)

fimalgoritmo

Qual será o resultado da execução desse algoritmo?

Atividade



1. Explique, com suas palavras, o processo de armazenamento de dados na memória de um computador.
2. Defina o conceito de variável e explique o procedimento para a declaração de uma variável. Cite pelo menos três exemplos que demonstrem esse procedimento.
3. Defina o conceito de constante e cite exemplos de atribuição de valores constantes às variáveis.

Atividade



4. Quais as palavras reservadas para os tipos de dados que estudamos nesta aula?
5. Qual é o único símbolo permitido no identificador de uma variável?
6. Por que os algoritmos lidam com diferentes tipos de dados?
7. Quais são os tipos de dados manipulados pelos algoritmos? Cite exemplos de cada um deles.

Atividade



8. Escreva um algoritmo que, dado o valor unitário da mercadoria e a quantidade comprada, exiba o valor total a ser pago.
9. Escreva um algoritmo que, dado um número, exiba o seu antecessor.
10. Escreva um algoritmo que, dado dois números, retorne se eles são iguais.

Referências



- *Salvetti, D. D.; Barbosa, L. M. Algoritmos. São Paulo: Makron Books, 2004. 300p;*
- *Forbellone, A. L. V.; Eberspächer, H. F. **Lógica de Programação – A construção de algoritmos e estruturas de dados.** 3.ed. São Paulo: Makron Books, 2005.*
- *Manzano, J. A; Oliveira, J. F. **Algoritmos - Estudo dirigido.** 15ª edição revisada. Editora Érica, 2012.*
- Instituto Metrópole Digital (IMD), Material Didático, UFRN, 2012.