



UNIVERSIDADE FEDERAL DE GOIÁS
INSTITUTO DE INFORMÁTICA



Introdução à Computação

Prof. Daniel Ventura



Introdução à Computação

- Construção de Algoritmos: elementos fundamentais



Estrutura Sequencial

Algoritmo

início
<tipo: nome da variável>;
<bloco de comandos>
fim.

Algoritmo em C

```
int main(void){  
  <tipo nome da variável>;  
  <bloco de comandos>  
}
```



Tipos Primitivos de Dados

Temos 4 tipos básicos de dados:

- Inteiro
- Real
- Caracter
- Lógico



Tipos Primitivos

- **INTEIRO**: Podem ser positivos, negativos ou nulos e não possuem parte decimal.

- (a) Eu tenho 2 filhos
- (b) A escada possui 8 degraus
- (c) Meu tio comprou 2 motos

Observação: 8 no item (b) é um dado do tipo inteiro e a informação é associar que 8 é o número de degraus da escada.



Tipos Primitivos

- **REAL**: Podem ser positivos, negativos ou nulos e possuem parte decimal. Possuem de 11 a 12 dígitos significativos com sinal.

- (a) Ele pesa 80,5 kg.
- (b) Tenho R\$ 654,40 na poupança.
- (c) A taxa de juros é de 5%.

Observação: Em C, a separação entre a parte inteira e a parte decimal de um número é feita pelo ponto (.).



Tipos Primitivos

- **CARACTER/STRING**: São dados formados por um único caractere ou por uma cadeia deles.

Esses caracteres podem ser as letras maiúsculas ou minúsculas, os números e os caracteres especiais (&, #, @, ?, +).

- (a) Constava na prova: "Use somente caneta".
- (b) O parque municipal estava repleto de placas:
"Não pise na grama!".
- (c) O nome do vendedor é "Felisberto Laranjeira".

Observação: Em C, os dados do tipo caracter são representados entre apóstrofes.



Tipos Primitivos

- **LÓGICO**: São chamados dados booleanos e são representados pelas palavras **true** e **false**, isto é, verdadeiro e falso.

Possuem apenas duas possibilidades de representação, ou seja, um dado do tipo lógico pode assumir apenas duas situações.

- (a) A porta pode estar *aberta* ou *fechada*.
- (b) A lâmpada pode estar *acesa* ou *apagada*.
- (c) O valor de x é igual ou diferente de 0.
- (d) A nota na disciplina é maior ou igual a 6 ou é menor do que 6.



Constantes

Um dado constante é o que não sofre nenhuma variação no decorrer do tempo. Seu valor é constante do início ao fim da execução do algoritmo.

Algoritmo	C
constante nome da constante = valor;	#define NOME_DA_CONSTANTE valor;
Ex: constante pi = 3,14;	Ex: #define PI 3.14;

Observação: Definições das constantes são feitas antes das declarações das variáveis.



Variáveis

Um dado é classificado como variável quando tem a **possibilidade de ser alterado** em algum instante no decorrer do tempo, ou seja, **durante a execução do algoritmo** em que é utilizado, o valor do dado sofre alteração.

Algoritmo	C
tipo: lista de variáveis;	tipo lista de variáveis;
Ex: inteiro: x, x1, a; real: abc, y;	Ex: int x, x1, a; float abc, y;



Formação de identificadores

- identificadores** são os **nomes** das variáveis, dos programas, das constantes, das rotinas/funções e tipos.

Ex:

constante pi = 3,14;
real: raio, area;



Regras para a formação

- Devem começar por um caracter alfabético;
- Podem ser seguidos por mais caracteres alfabéticos ou numéricos;
- Não devem ser usados caracteres especiais (incluindo o espaço em branco);
- Não podemos usar palavras reservadas nos identificadores, que são nomes utilizados pelo compilador para representar comandos de controle do programa (início, fim, se, ...)

Exemplos:

válidos: Alpha, X, BJ153, K7, Notas, ABC.

não-válidos: 5X, E(13), A:B, Nota/2; AWQ*, P&AA.



Tipos de dados em C (Exemplos)

char	um único byte
int	16 bits
short	16 bits
long	32 bits
long long	64 bits
float	32 bits
double	64 bits



Exercícios

1. Identifique o tipo dos dados:

a – inteiro	b – real	c – lógico	d – caracter/string.
() true	() 45,0	() 1234	() 0,0
() "aula"	() "c*d"	() -234	() "1 2 3 4"
() -0,342	() 35,23	() "34"	() -18,589
() ' '	() -354,0	() -15,0	() false
() 0	() 897	() "false"	() -22

2. Assinale os identificadores válidos e, para cada identificador não assinalado, explique a razão de ele ser inválido.

a) (X)	b) U2	c) AH!	d) 'ALUNO'	e) #55
f) KM/L	g) UYT	h) ASDRUBAL	i) AB°C	j) O&O
l) P{O}	m) B52	n) Rua	o) CEP	p) dia/mês



Exercícios

3. Supondo que as variáveis Na, Nm, NMat, Sx sejam utilizadas para armazenar a nota do aluno, nome do aluno, o número da matrícula e o sexo, declare-as corretamente na linguagem algorítmica, associando o tipo primitivo adequado ao dado que será armazenado.

4. Encontre os erros da seguinte declaração de variáveis:

inteiro: Endereco, NFilhos;

real: Xpto, c, Peso, R\$;

logico: Lampada, c;



Expressões Aritméticas

- As **expressões aritméticas** são escritas linearmente usando a notação matemática, tendo como resposta sempre um valor numérico.
- Seus **operandos** são constantes ou variáveis do tipo numérico.
- Operadores aritméticos** são o conjunto de símbolos que representa as operações básicas da matemática.
- Algumas operações serão representadas através de palavras-chave ao invés de símbolos (função).



Expressões Aritméticas

Operador	Função	Operandos	Resultado
+	Adição	Z ou R	Z ou R
-	Subtração	Z ou R	Z ou R
*	Multiplicação	Z ou R	Z ou R
/	Divisão	Z ou R	R
div	Quociente da divisão	Z	Z
mod	Resto da divisão	Z	Z

Ex: $2+3$, $n-m$, $6/2$, $9 \bmod 2$, $24 \text{ div } 5$.



Expressões Aritméticas

Operador	Função	Operandos	Resultado
pot(x,y)	Potenciação	Z ou R	Z ou R
rad(x)	Radiciação	Z ou R	R

Ex: $\text{pot}(2,4)$, $\text{rad}(81)$.

Prioridades de Operadores:

1ª - ()

2ª – pot, rad

3ª - * , / , div , mod

4ª - + , -



Expressões Aritméticas

Operadores aritméticos em C

Operador	Função	Operandos	Resultado
+	Adição	Z ou R	Z ou R
-	Subtração	Z ou R	Z ou R
*	Multiplicação	Z ou R	Z ou R
/	Divisão	Z ou R	Z ou R
%	Resto da divisão	Z	Z
pow(x,y)	Potenciação	Z ou R	R
sqrt(x)	Radiciação	Z ou R	R



Expressões Lógicas

- As **expressões lógicas** são expressões que trabalham com operadores relacionais ou lógicos, tendo como resposta um valor lógico. (**true** ou **false**)
- Uma **relação** é uma comparação entre valores do mesmo tipo.
- Os **operadores lógicos** permitem a formação de novas proposições lógicas a partir de outras mais simples



Expressões Lógicas

Operadores Relacionais

Operador	Função
=	Igual a
>	Maior que
<	Menor que
>=	Maior ou igual a
<=	Menor ou igual a
<>	Diferente de

Ex: $2 * 6 = 36 / 3$
 $12 = 12$
 V



Expressões Lógicas

Operadores Relacionais em C

Operador	Função
==	Igual a
>	Maior que
<	Menor que
>=	Maior ou igual a
<=	Menor ou igual a
!=	Diferente de



Expressões Lógicas

Operadores Lógicos

Operador	Função
não	negação
e	conjunção
ou	disjunção

Ex: Se chover **e** relampejar, eu fico em casa
 Se chover **ou** relampejar, eu fico em casa

OBS: temos um 'ou' inclusivo ao invés do 'ou' exclusivo da linguagem natural.



Tabelas-verdade

A	B	A ou B
V	V	V
V	F	V
F	V	V
F	F	F

A	B	A e B
V	V	V
V	F	F
F	V	F
F	F	F

A	não A
V	F
F	V



Expressões Lógicas

Operadores Lógicos em C

Operador	Função
!	negação
&&	conjunção
	disjunção



Expressões Lógicas

Prioridades entre operadores lógicos:

1ª - **não**

2ª - **e**

3ª - **ou**

Prioridade para operadores:

1ª - ()

2ª - operadores aritméticos

3ª - operadores relacionais

4ª - operadores lógicos

OBS: Em caso de empate entre as prioridades, resolver da esquerda para a direita (associatividade à esquerda).



Exercícios

1. Determine os resultados obtidos na avaliação das expressões lógicas seguintes:

a. $x * x + y > z$

b. $x * x + y > z$

c. $(\text{Nome} = \text{"JORGE"}) \text{ e Sim ou } (\text{rad}(x) < z * 10)$

d. $(\text{rad}(x) > \text{rad}(y)) \text{ ou não Sim e } (\text{Nome} = \text{"PEDRO"})$

Considerando que os valores iniciais das variáveis sejam:

a. $x = 1, y = 2, z = 5$

b. $x = 4, y = 3, z = 1$

c. $x = 2, z = 9, \text{Nome} = \text{"MARIA"}, \text{Sim} = \text{false}$

d. $x = 4, y = 9, \text{Sim} = \text{true}, \text{Nome} = \text{"ANA"}$

