

INTRODUÇÃO À PROGRAMAÇÃO

Aula 23

Operadores lógicos e funções lógicas

A operação OR $A \mid B$ retorna 1s onde pelo menos um dos arranjos A ou B tiver elementos diferentes de zero.

```
>> z = 0|3
```

```
z =
```

```
logical
```

```
1
```

```
>> z = 0|0
```

```
z =
```

```
logical
```

```
0
```

Operadores lógicos e funções lógicas

A operação OR $A \mid B$ retorna 1s onde pelo menos um dos arranjos A ou B tiver elementos diferentes de zero.

```
>> clear
```

```
>> z = [5, -3, 0, 0] | [2, 4, 0, 5]
```

```
z =
```

1×4 logical array

```
1 1 0 1
```

Operadores lógicos e funções lógicas

A operação OR $A \mid B$ retorna 1s onde pelo menos um dos arranjos A ou B tiver elementos diferentes de zero.

```
>> z = 3 < 5 | 4 == 7
```

Operadores lógicos e funções lógicas

A operação OR $A \mid B$ retorna 1s onde pelo menos um dos arranjos A ou B tiver elementos diferentes de zero.

```
>> z = 3 < 5 | 4 == 7
```

```
z =
```

```
logical
```

```
1
```

```
>> z = (3 < 5) | (4 == 7)
```

```
z =
```

```
logical
```

```
1
```

Operadores lógicos e funções lógicas

A operação OR $A \mid B$ retorna 1s onde pelo menos um dos arranjos A ou B tiver elementos diferentes de zero.

```
>> z = ~3 == 7 | 4 == 6
```

Operadores lógicos e funções lógicas

A operação OR $A \mid B$ retorna 1s onde pelo menos um dos arranjos A ou B tiver elementos diferentes de zero.

```
>> z = ~3 == 7 | 4 == 6
```

```
z =
```

```
logical
```

```
0
```

```
>> z = ((~3) == 7) | (4 == 6)
```

```
z =
```

```
logical
```

```
0
```

Operadores lógicos e funções lógicas

A função OU exclusivo (exclusive OR) ou xor (A,B):

- ” Retorna 0 onde A e B forem ambos diferentes de zero.
- ” Retorna 1 onde A ou B forem diferentes de zero, mas não ambos.
- ” operação OR $A \mid B$ retorna 1s onde pelo menos um dos arranjos A ou B tiver elementos diferentes de zero.

```
>> z = xor([3,0,6],[5,0,0])
```

z =

1×3 logical array

0 0 1

```
>> z = [3,0,6] |[5,0,0]
```

z =

1×3 logical array

1 0 1

Operadores lógicos e funções lógicas

Os operadores de curto-circuito AND (&&) e OR (||):

- “ Realizam operações somente em escalares
- “ São chamados desta forma, porque avaliam o segundo operando apenas quando o resultado não estiver definido pelo primeiro operando
- “ $A \&\& B$, retorna verdadeiro (1 lógico) se ambos A e B forem verdadeiros, e falso (0 lógico) caso contrário.
- “ $A || B$, retorna verdadeiro (1 lógico) se A ou B ou ambos forem verdadeiros, e falso (0 lógico) caso contrário.

Operadores lógicos e funções lógicas

```
>> 1 & 0
```

```
ans =
```

```
logical
```

```
0
```

```
>> 1 && 0
```

```
ans =
```

```
logical
```

```
0
```

```
>> [1,1] & [0,0]
```

```
ans =
```

```
1×2 logical array
```

```
0 0
```

```
>> [1,1] && [0,0]
```

Operands to the `||` and `&&` operators must be convertible to logical scalar values.

Operadores lógicos e funções lógicas

```
>> 1 | 0
```

```
ans =
```

```
logical
```

```
1
```

```
>> [1,1] | [0,1]
```

```
ans =
```

```
1×2 logical array
```

```
1 1
```

```
>> 1 || 0
```

```
ans =
```

```
logical
```

```
1
```

```
>> [1,1] || [0,1]
```

Operands to the `||` and `&&` operators must be convertible to logical scalar values.

Operadores lógicos e funções lógicas

TABELA 4.3-4 Funções lógicas

Função lógica	Definição
<code>all (x)</code>	Retorna um escalar, que é 1 se todos os elementos no vetor <code>x</code> forem diferentes de zero, ou 0 caso contrário.
<code>all (A)</code>	Retorna um vetor linha com o mesmo número de colunas que a matriz <code>A</code> contendo 1s e 0s, dependendo de a coluna correspondente de <code>A</code> ter ou não todos os elementos diferentes de zero.
<code>any (x)</code>	Retorna um escalar, que é 1 se qualquer um dos elementos no vetor <code>x</code> for diferente de zero, ou 0 caso contrário.
<code>any (A)</code>	Retorna um vetor linha com o mesmo número de colunas que a matriz <code>A</code> contendo 1s e 0s, dependendo de a coluna correspondente da matriz <code>A</code> conter ou não algum elemento diferente de zero.

Operadores lógicos e funções lógicas

<code>find(A)</code>	Gera um arranjo que contém os índices dos elementos do arranjo A diferentes de 0.
<code>[u,v,w] = find(A)</code>	Gera os arranjos u e v que contém os índices de linha e de coluna dos elementos diferentes de zero do arranjo A e gera o arranjo w que contém os valores dos elementos diferentes de zero. O arranjo w pode ser omitido.
<code>finite(A)</code>	Retorna um arranjo do mesmo tamanho de A com 1s onde os elementos são finitos e 0s nas outras posições.
<code>ischar(A)</code>	Retorna 1, se A for um arranjo de caracteres, ou 0, caso contrário.
<code>isempty(A)</code>	Retorna 1, se A for uma matriz vazia, ou 0, caso contrário.
<code>isinf(A)</code>	Retorna um arranjo do mesmo tamanho de A com 1s onde os elementos são Inf e 0s nas outras posições.
<code>isnan(A)</code>	Retorna um arranjo do mesmo tamanho de A com 1s onde os elementos são NaN e 0s nas outras posições. (NaN representa “não é um número” (<i>Not a Number</i>), o que significa um resultado indefinido.)
<code>isnumeric(A)</code>	Retorna 1, se A for um arranjo numérico, ou 0, caso contrário.
<code>isreal(A)</code>	Retorna 1, se A não tiver elementos com parte imaginária, ou 0, caso contrário.
<code>logical(A)</code>	Converte os elementos de um arranjo A em valores lógicos.
<code>xor(A,B)</code>	Retorna um arranjo com a mesma dimensão de A e B; o novo arranjo terá 1s onde A ou B forem diferentes de zero, mas não ambos, e 0s onde A e B forem ambos diferentes de zero ou iguais a zero.
