

# Introdução à Programação

## Capítulo 6 - Operadores Lógicos

# Operações Lógicas

- Compreendem a base para a construção de sistemas digitais e da lógica proposicional
- Estas operações ajudam na construção de instruções em que há a tomada de decisões (Estruturas de Decisões)
- Para auxiliar no entendimento do resultado dos operadores lógicos, utilizamos a **Tabela Verdade**

# Tabela Verdade

a	b	a E b	a OU b	Não a	Não b
true	true	true	true	false	false
true	false	false	true	false	true
false	true	false	true	true	false
false	false	false	false	true	true

# Tabela Verdade

- E: só será `true` se todas as sentenças forem `true`
- OU: só será `false` se todas as sentenças forem `false`
- Não: inverte a sentença

# Operadores Lógicos

Nome	Operador	Utilização	Operação
E (AND)	&&	expr1 && expr2	Retorna <code>expr1</code> caso possa ser convertido para falso; senão retorna <code>expr2</code>
OU (OR)		expr1    expr2	Retorna <code>expr1</code> caso possa ser convertido para verdadeiro; senão retorna <code>expr2</code>
Não (Not)	!	!expr	Retorna falso caso possa ser convertido para verdadeiro; senão retorna verdadeiro

# Operadores Lógicos

- Exemplos

```
true && true;      // t && t retorna true
true && false;     // t && f retorna false
false && true;     // f && t retorna false
false && (3 == 4); // f && f retorna false
"Gato" && "Cão";   // t && t retorna "Cão"
false && "Gato";   // f && t retorna false
"Gato" && false;   // t && f retorna false
```

# Operadores Lógicos

- Exemplos

```
true || true;      // t || t retorna true
false || true;     // f || t retorna true
true || false;     // t || f retorna true
false || (3 == 4); // f || f retorna false
"Gato" || "Cão";   // t || t retorna "Gato"
false || "Gato";   // f || t retorna "Gato"
"Gato" || false;   // t || f retorna "Gato"
```

# Operadores Lógicos

- Exemplos

```
!true;    // !t retorna false  
!false;   // !f retorna true  
!"Gato";  // !t retorna false
```



# Operadores Lógicos

```
var nota1, nota2, faltas, media;
nota1 = parseFloat(prompt("Digite a 1a nota"));
nota2 = parseFloat(prompt("Digite a 2a nota"));
faltas = parseInt(prompt("Digite a quantidade de faltas"));
media = (nota1 + nota2) / 2;
if (faltas <= 15) {
    if (media >= 6.0) {
        document.write("O estudante esta aprovado");
    } else {
        document.write("O estudante esta reprovado");
    }
} else {
    document.write("O estudante esta reprovado por falta");
}
```

# Operadores Lógicos

```
var nota1, nota2, faltas, media;
nota1 = parseFloat(prompt("Digite a 1a nota"));
nota2 = parseFloat(prompt("Digite a 2a nota"));
faltas = parseInt(prompt("Digite a quantidade de faltas"));
media = (nota1 + nota2) / 2;
if (faltas <= 15 && media >= 6.0) {
    document.write("O estudante esta aprovado");
} else if (faltas <= 15 && media < 6.0) {
    document.write("O estudante esta reprovado");
} else {
    document.write("O estudante esta reprovado por falta");
}
```

# Avaliação de Curto-Circuito

- Como expressões lógicas são avaliadas da esquerda para a direita, elas são testadas como possíveis avaliações de "curto-circuito" utilizando as seguintes regras:
  - `false && qualquercoisa` é avaliado em curto-circuito como falso
  - `true || qualquercoisa` é avaliado em curto-circuito como verdadeiro

Repare que a parte `qualquercoisa` das expressões acima não é avaliada, de forma que qualquer efeito colateral de fazê-lo não produz efeito algum.

# Avaliação de Curto-Circuito

- Exemplos

```
var a = 2, b = 1, c = 0, d = -1;  
  
a > 2 && b < 1 && c > -1 && d < -2 && a < b && d >= c // false  
  
a >= 2 || b > 1 || d > -2 || a < d || a > b || d >= c // true
```