

---

# Algoritmos e Lógica de Programação

Introdução a Linguagem C

Condicionais

---

Prof. Rhafael Freitas da Costa

# Conteúdo

- Retomada dos principais pontos da aula anterior;
- Estruturas condicionais simples, compostas e aninhadas (SE, ESCOLHA);
- Continuidade da introdução à programação em linguagem C;
- Ao final do componente curricular será possível utilizar estruturas condicionais simples e compostas para resolver problemas algorítmicos.

---

---

# Revisão

— Na aula anterior... —

---

---

# Estruturas condicionais

- Estruturas condicionais ou estruturas de decisão permitem a escolha de um conjunto de instruções a serem executadas diante da avaliação de uma condição (expressão lógica);
- Qualquer condição (expressão lógica) pode ser utilizada, desde que o resultado seja VERDADEIRO ou FALSO;
  - Utilizar expressões com operadores relacionais, lógicos ou variáveis do tipo lógico 3.

# Estruturas condicionais – SE / IF

```
if (<condição>) {  
    <bloco de comandos VERDADEIRO>  
}
```

Se a <condição> for **VERDADEIRA**, será executado o

<bloco de comandos VERDADEIRO>

Se a <condição> for **FALSA**, nada será feito

**ATENÇÃO** para o uso de **{ }**

# Estruturas condicionais – SE / IF

## //Portugol WebStudio

```
programa {  
    funcao inicio() {  
        real valorA, valorB, result  
  
        escreva("Digite o valorA: ")  
        leia(valorA)  
  
        escreva("Digite o valorB: ")  
        leia(valorB)  
  
        se (valorB != 0) {  
            result = valorA / valorB  
            escreva("Resultado da divisao: ", result)  
        }  
    }  
}
```

```
Digite o valorA: 5.9  
Digite o valorB: 6.1  
Resultado da divisao: 0.967213
```

```
Digite o valorA: 7.3  
Digite o valorB: 0  
Resultado da divisao: 0
```

## // Linguagem C

```
#include <stdio.h>  
int main() {  
    float valorA, valorB, result;  
  
    printf("Digite o valorA: ");  
    scanf("%f", &valorA);  
  
    printf("Digite o valorB: ");  
    scanf("%f", &valorB);  
  
    if (valorB != 0) {  
        result = valorA / valorB;  
        printf("Resultado da divisao: %f", result);  
    }  
  
    return 0;  
}
```

# Estruturas condicionais – SE-SENAO / IF-ELSE

```
if (<condição>) {  
    <bloco de comandos VERDADEIRO>  
} else {  
    <bloco de comandos FALSO>  
}
```

Se a <condição> for **VERDADEIRA**, será executado o

<bloco de comandos VERDADEIRO>,

Se a <condição> for **FALSA**, será executado o

<bloco de comandos FALSO>,

**ATENÇÃO!** O **else** não tem condição

# Estruturas condicionais – SE / IF

## //PortugolWebStudio

```
programa {
    funcao inicio() {
        real valorA, valorB, result

        escreva("Digite o valorA: ")
        leia(valorA)

        escreva("Digite o valorB: ")
        leia(valorB)

        se (valorB != 0) {
            result = valorA / valorB
            escreva("Resultado da divisao: ", result)
        } senão {
            escreva("Impossivel dividir")
        }
    }
}
```

## // Linguagem C

```
#include <stdio.h>
int main() {
    float valorA, valorB, result;

    printf("Digite o valorA: ");
    scanf("%f", &valorA);

    printf("Digite o valorB: ");
    scanf("%f", &valorB);

    if (valorB != 0) {
        result = valorA / valorB;
        printf("Resultado da divisao: %f", result);
    } else {
        printf("Impossivel dividir!");
    }
    return 0;
}
```



# Estruturas condicionais – SE-SENAO SE-SENAO / IF-ELSE IF-ELSE



# Estruturas condicionais – SE-SENAO SE-SENAO / IF-ELSE IF-ELSE

```
// Linguagem C

#include <stdio.h>

int main() {
    float media;

    printf("Digite a media:");
    scanf("%f", &media);

    if (media >= 9.0) {
        printf("Conceito A");
    }

    if ((media < 9) && (media >= 7.5)) {
```

→ *(segue no quadro ao lado)*

```
        if ((media < 7.5) && (media >= 6.0)) {
            printf("Conceito C");
        }

        if (media < 6.0) {
            printf("Conceito D");
        }

        return 0;
    }
```

# Estruturas condicionais – SE-SENAO SE-SENAO / IF-ELSE IF-ELSE

```
// Linguagem C

#include <stdio.h>

int main() {
    float media;

    printf("Digite a media:")
    scanf("%f", &media);

    if (media >= 9.0) {
        printf("Conceito A");
    }

    if ((media < 9) && (media >= 7.5)) {
        printf("Conceito B");
    }
}
```

→ (segue no quadro ao lado)

```
if ((media < 7.5) && (media >= 6.0)) {
    printf("Conceito C");
}
```

```
if (media < 6.0) {
    printf("Conceito D");
}
```

Não é a solução mais adequada porque  
obrigatoriamente todas as condições serão  
testadas

# Estruturas condicionais – SE-SENAO SE-SENAO / IF-ELSE IF-ELSE

```
if (<condição>) {  
    <bloco de comandos VERDADEIRO>  
} else if (<condição2>) {  
    <bloco de comandos VERDADEIRO>  
} else {  
    <bloco de comandos FALSO>  
}
```

**ATENÇÃO!** O **else** não tem condição

Se a <condição1> for **VERDADEIRA**, será executado o

<bloco de comandos VERDADEIRO1>

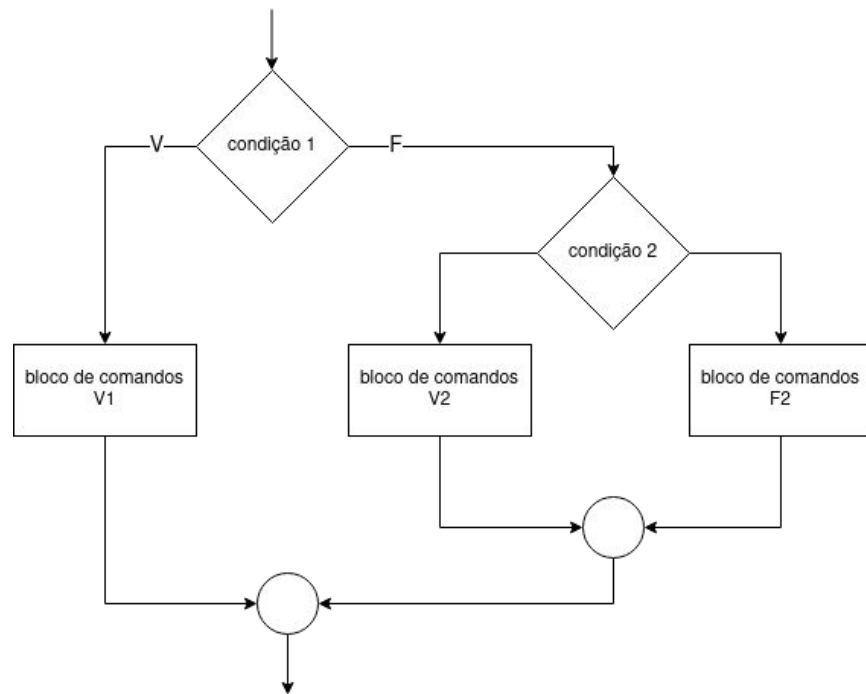
Se a <condição1> for **FALSA**, será avaliada a <condição2>

Se a <condição2> for **VERDADEIRA**, será executado o

<bloco de comandos VERDADEIRO2>

Se a <condição2> for **FALSA**, será executado o <bloco de comandos FALSO>

# Estruturas condicionais – SE-SENAO SE-SENAO / IF-ELSE IF-ELSE



# Estruturas condicionais – SE-SENAO SE-SENAO / IF-ELSE IF-ELSE

## // Portugol WebStudio

```
programa {
funcao inicio() {

    inteiro idade

    escreva("Digite a idade: ")
    leia(idade)

    se (idade < 16) {
        escreva("Nao pode votar")
    } senao se ((idade >= 16) e (idade < 18)) {
        escreva("Voto facultativo")
    } senao se ((idade >= 18) e (idade < 70)) {
        escreva("Voto obrigatorio")
    } senao {
        escreva("Voto facultativo")
    }
}
```

## // Linguagem C

```
#include <stdio.h>

int main() {

    int idade;



    printf("Digite a idade: ");
    scanf("%d", &idade);

    if (idade < 16) {
        printf("Nao pode votar");
    } else if ((idade >= 16) e (idade < 18)) {
        printf("Voto facultativo");
    } else if ((idade >= 18) e (idade < 70)) {
        printf("Voto obrigatorio");
    } else {
        printf("Voto facultativo");
    }



    return 0;
}
```

# Estruturas condicionais – SE-SENAO SE-SENAO / IF-ELSE IF-ELSE

// Portugol WebStudio

```
programa {  
  funcao inicio() {  
  
    inteiro idade  
  
    escreva("Digite a idade: ")  
    leia(idade)  
  
    se (idade < 16) {  
      escreva("Nao pode votar")  
    } senao se ((idade >= 16)   
      escreva("Voto facultativo")  
    } senao se ((idade >= 18)   
      escreva("Voto obrigatorio")  
    } senao {  
      escreva("Voto facultativo")  
    }  
  }  
}
```

// Linguagem C

```
#include <stdio.h>  
  
int main() {  
    int idade;  
  
    de: ");  
  
    votar");  
    16)  (idade < 18)) {  
        ltativo");  
    18)  (idade < 70)) {  
        gatorio");  
    } else {  
        printf("Voto facultativo");  
    }  
  
    return 0;  
}
```

Temos duas faixas etárias com voto facultativo.  
Posso colocar as duas na mesma condição (expressão lógica)?

# Estruturas condicionais – SE-SENAO SE-SENAO / IF-ELSE IF-ELSE

// Portugol WebStudio

```
programa {
    funcao inicio() {
        inteiro idade

        escreva("Digite a idade: ")
        leia(idade)

        se ((idade >= 18) e (idade < 70)) {
            escreva("Voto obrigatorio")
        }
        senao se (((idade >= 16) e (idade < 18))ou (idade
>= 70)) {
            escreva("Voto facultativo")
        } senao {
            escreva("Nao pode votar")
        }
    }
}
```

// Linguagem C

```
#include <stdio.h>

int main() {
    int idade;

    printf("Digite a idade: ");
    scanf("%d", &idade);

    if ((idade >= 18) && (idade < 70)) {
        printf("Voto obrigatorio");
    }
    else if (((idade >= 16) && (idade < 18))|| (idade >= 70)){
        printf("Voto facultativo");
    }
    else {
        printf("Nao pode votar");
    }
    return 0;
}
```



# Estruturas condicionais – Síntese

ATENÇÃO para o uso de {}. O **else** é opcional e não tem condição.

```
if (<condição>) {  
    <bloco de comandos>  
}
```

```
if (<condição>) {  
    <bloco de comandos>  
} else {  
    <bloco de comandos>  
}
```

```
if (<condição1>) {  
    <bloco de comandos>  
} else if (<condição2>) {  
    <bloco de comandos>  
}
```

```
if (<condição1>) {  
    <bloco de comandos>  
} else if (<condição2>) {  
    <bloco de comandos>  
} else if (<condição3>) {  
    <bloco de comandos>  
} else {  
    <bloco de comandos>  
}
```

# Estruturas condicionais – Síntese

ATENÇÃO para o uso de **{}**. O **senao** é opcional e não tem condição.

```
se (<condição>) {  
    <bloco de comandos>  
}
```

```
se (<condição>) {  
    <bloco de comandos>  
} senao {  
    <bloco de comandos>  
}
```

```
se (<condição1>) {  
    <bloco de comandos>  
} senao se (<condição2>) {  
    <bloco de comandos>  
}
```

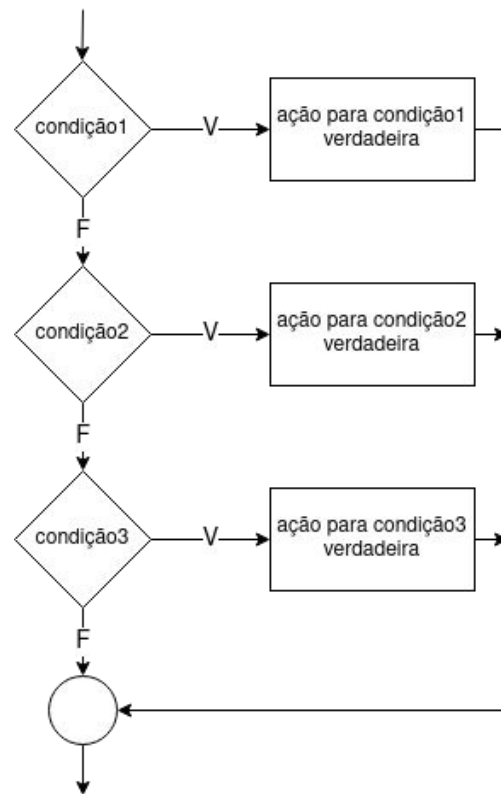
```
se (<condição1>) {  
    <bloco de comandos>  
} senao se (<condição2>) {  
    <bloco de comandos>  
} senao se (<condição3>) {  
    <bloco de comandos>  
} senao {  
    <bloco de comandos>  
}
```

## Estruturas condicionais – ESCOLHA-CASO / SWITCH-CASE

- Apenas um entre diferentes blocos de comandos pode ser executado de acordo com o valor de uma variável de teste;
- Alternativa mais rápida ao uso de tomadas de decisão sequenciais em que é necessário testar uma mesma variável diversas vezes;
- **A variável de teste deve ser do tipo inteiro ou caractere.**

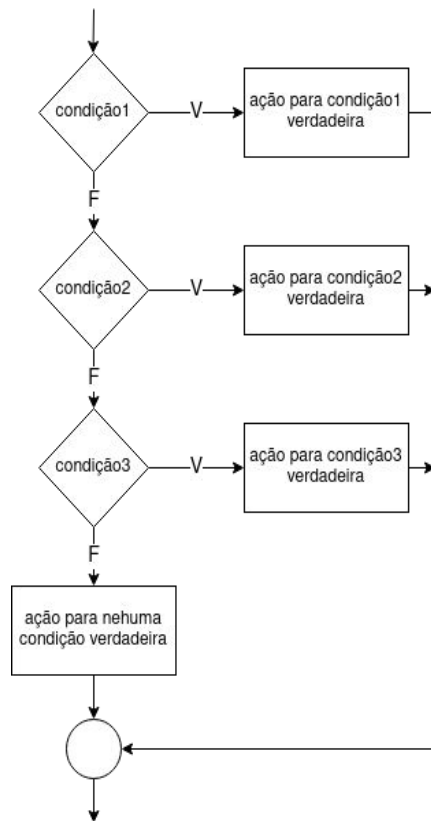
# Estruturas condicionais – ESCOLHA-CASO / SWITCH-CASE

```
escolha (<variavel>) {  
  caso <valor1>:  
    <bloco de comandos 1>  
  pare  
  caso <valor2>:  
    <bloco de comandos 2>  
  pare  
  ...  
  caso <valorN>:  
    <bloco de comandos N>  
  pare  
}
```



# Estruturas condicionais – ESCOLHA-CASO / SWITCH-CASE

```
escolha (<variavel>) {  
    caso <valor1>:  
        <bloco de comandos 1>  
    pare  
    caso <valor2>:  
        <bloco de comandos 2>  
    pare  
    caso contrario:  
        <bloco de comandos para nenhum caso verdadeiro>  
}
```



# Estruturas condicionais – ESCOLHA-CASO / SWITCH-CASE

```
se (<condição1>) {  
    <bloco de comandos VERDADEIRO1>  
} senao se (<condição2>) {  
    <bloco de comandos VERDADEIRO2>  
} senao se (<condição3>) {  
    <bloco de comandos VERDADEIRO3>  
} senao se {  
    ...  
}
```

```
escolha (<variavel>) {  
    caso <valor1>:  
        <bloco de comandos 1>  
    pare  
    caso <valor2>:  
        <bloco de comandos 2>  
    pare  
    caso <valorN>:  
        <bloco de comandos N>  
    pare  
}
```

# Estruturas condicionais – ESCOLHA-CASO / SWITCH-CASE

```
--- Cafeteria da UP ---  
Digite o codigo do produto de 1 a 6: 4  
Pastel  
Programa finalizado.
```

```
    escreva("--- Cafeteria da UP ---\n")  
    escreva("Digite o codigo do produto de 1 a  
6: ")  
    leia(codigo)
```

```
escolha(codigo) {
```

```
    caso 1:
```

```
        escreva("Cafe preto\n")
```

```
        pare
```

```
    caso 2:
```

```
        escreva("Cafe com leite")
```

```
        pare
```

```
    caso 3:
```

```
        escreva("Chocolate quente")
```

```
        pare
```

```
--- Cafeteria da UP ---  
Digite o codigo do produto de 1 a 6: 6  
Coxinha  
Programa finalizado.
```

```
    caso 4:
```

```
        escreva("Pastel")
```

```
        pare
```

```
    caso 5:
```

```
        escreva("Cachorro-quente")
```

```
        pare
```

```
    caso 6:
```

```
        escreva("Coxinha")
```

```
    default:
```

```
        escreva("Codigo invalido")
```

```
    }
```

```
}
```

```
}
```

```
--- Cafeteria da UP ---  
Digite o codigo do produto de 1 a 6: 9  
Codigo invalido  
Programa finalizado.
```

# Estruturas condicionais – ESCOLHA-CASO / SWITCH-CASE

```
programa {
    funcao inicio() {
        inteiro codigo
        escreva("--- Cafeteria da UP ---\n")
        escreva("Digite o codigo do produto de 1 a
6: ")
        leia(codigo)

        escolha(codigo) {
            caso 1:
                escreva("Cafe preto\n")
                pare
            caso 2:
                escreva("Cafe com leite")
                pare
            caso 3:
                escreva("Chocolate quente")
                pare
            caso 4:
                escreva("Pastel")
                pare
            caso 5:
                escreva("Cachorro-quente")
                pare
            caso 6:
                escreva("Coxinha")
                pare
            caso contrario:
                escreva("Codigo invalido")
        }
    }
}
```



# Estruturas condicionais – ESCOLHA-CASO / SWITCH-CASE

```
switch (<variavel>) {  
    case <rotulo1>:  
        <bloco de comandos 1>;  
        break;  
    case <rotulo2>:  
        <bloco de comandos 2>;  
        break;  
    ...  
    case <rotuloN>:  
        <bloco de comandos N>;  
        break;  
    default:  
        <bloco de comandos padrão>;  
        break;  
}
```

## ATENÇÃO!

Cada **case** deve  
obrigatoriamente terminar com

**; break;**

O **default** é opcional

# Estruturas condicionais – ESCOLHA-CASO / SWITCH-CASE

// Linguagem C

```
#include <stdio.h>
int main() {

    int codigo;

    printf("Digite o codigo: ");
    scanf("%d", &codigo);

    switch (codigo) {
        case 1: printf("Cafe preto"); break;
        case 2: printf("Cafe com leite"); break;
        case 3: printf("Chocolate quente"); break;
        case 4: printf("Pastel"); break;
        case 5: printf("Cachorro-quente"); break;
        case 6: printf("Coxinha"); break;
        default: printf("Codigo invalido"); break;
    }
    return 0;
}
```

```
❯ ./main
Digite o codigo: 23
Codigo invalido❯
```

```
❯ ./main
Digite o codigo: 4
Pastel❯
```

```
❯ ./main
Digite o codigo: 1
Cafe preto❯
```

```
❯ ./main
Digite o codigo: 6
Coxinha❯
```

# Estruturas condicionais – ESCOLHA-CASO / SWITCH-CASE

- Usar apenas uma variável no comando **switch**;
- A variável deve ser do tipo **inteiro** ou **caracter**;
- O **case** executado corresponde ao valor exato da variável de teste;
- No máximo um **case** será executado. Pode haver situações em que **nenhum case** seja executado;
- O comando **default** é opcional. Se existir, ele deve estar na última linha.

## Para praticar

1. Escreva um programa que receba um caractere e verifique se ele é uma vogal ou uma consoante.
2. Faça um programa que leia um número inteiro e imprima se é par ou ímpar, usando o comando `switch`.
3. Escreva um programa que receba um número inteiro de 1 a 7 e exiba o dia da semana correspondente.

## Para praticar

4. Em uma loja de eletrodomésticos, as compras têm um **preço à vista**, ou acréscimo de **10 % para pagamentos em 2 vezes**, ou ainda, acréscimo de **20% para pagamento em 3 vezes**. O algoritmo deve pedir para o usuário digitar o valor da compra à vista e a opção de compra. O algoritmo deve exibir qual o valor final a ser pago.
5. Construa um algoritmo que funcione como uma calculadora simples. A calculadora deve realizar as operações de **soma (+)**, **subtração (-)**, **multiplicação (\*)** e **divisão (/)**. O usuário deve informar ao algoritmo dois operandos (dois números) e a operação. O algoritmo deve retornar o **resultado da operação** ou uma **mensagem de erro** caso o usuário solicite uma operação não definida.

---

---

# Estruturas Condicionais

— Simples, Compostas e Aninhadas —

---

---

# Estruturas condicionais

- **Estruturas condicionais** ou **estruturas de decisão** permitem a escolha de um conjunto de instruções a serem executadas quando determinada condição aconteça;
- Qualquer condição / expressão lógica pode ser utilizada, desde que o resultado seja verdadeiro ou falso:
  - Utilizar expressões com operadores relacionais, lógicos ou variáveis do tipo lógico

# Estrutura condicional simples

- Estrutura condicional **IF**

```
if (condição) {  
  
    <bloco de comandos>  
  
}
```

Caso o resultado da condição for **VERDADEIRO**, será executado o bloco de comandos que está dentro da estrutura.

Caso o resultado seja **FALSO**, a execução do programa ignora o bloco de comandos

**ATENÇÃO**

para o uso de 



# Estrutura condicional simples

// Portugol WebStudio

```
programa {  
    funcao inicio() {  
  
        inteiro idade  
  
        escreva("Digite a idade da pessoa: ")  
        leia(idade)  
  
        se (idade < 18) {  
            escreva("A pessoa eh menor de idade \n")  
        }  
  
        escreva("A idade digitada foi: ", idade)  
    }  
}
```

Digite a idade da pessoa: 15  
A pessoa eh menor de idade  
A idade digitada foi: 15

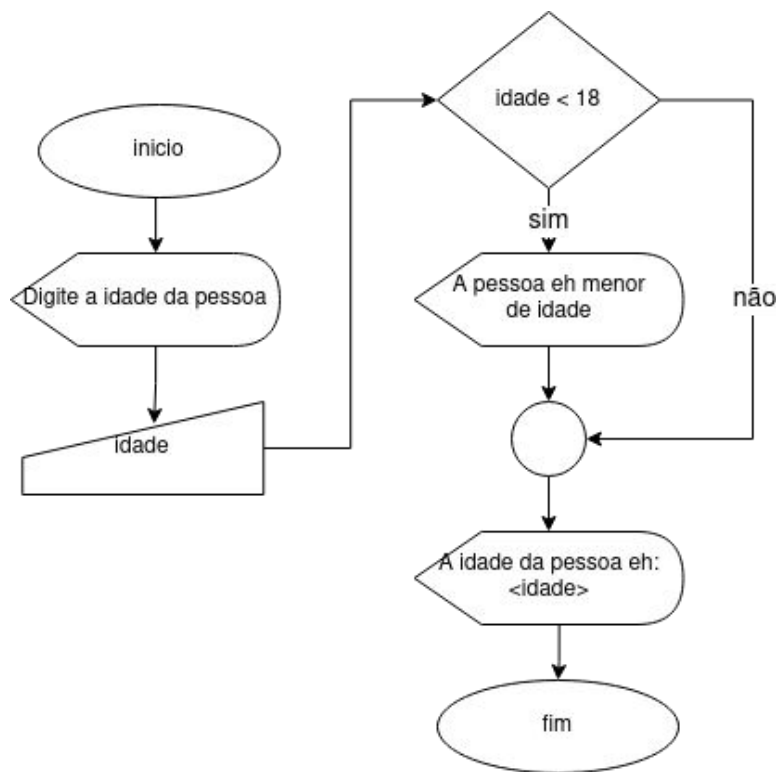
// Linguagem C

```
#include <stdio.h>  
  
int main() {  
  
    if (idade < 18) {  
        printf("A pessoa eh menor de idade \n");  
    }  
  
    printf("A idade digitada foi: %d", idade);  
  
    return 0;  
}
```

Digite a idade da pessoa: 21  
A idade digitada foi: 21

# Estrutura condicional simples

- Estrutura condicional **IF**



# Estrutura condicional composta

- Estrutura condicional **IF - ELSE**

```
if (<condição>) {  
    <bloco de comandos VERDADEIRO>  
} else {  
    <bloco de comandos FALSO>  
}
```

Caso o resultado da condição for **VERDADEIRO**, será executado o bloco de comandos VERDADEIRO.

Caso o resultado seja **FALSO**, será executado o bloco de comandos FALSO.

# Estrutura condicional composta

// Portugol WebStudio

```
programa {  
    funcao inicio() {  
        real notaA, notaB, media  
        escreva("Digite a notaA: ")  
        leia(notaA)  
        escreva("Digite a notaB: ")  
        leia(notaB)  
        media = (notaA + notaB) / 2  
        se (media >= 6.0) {  
            escreva("Aprovado")  
        }  
        senao {  
            escreva("Reprovado")  
        }  
    }  
}
```

```
Digite a notaA: 5.9  
Digite a notaB: 8.1  
Aprovado
```

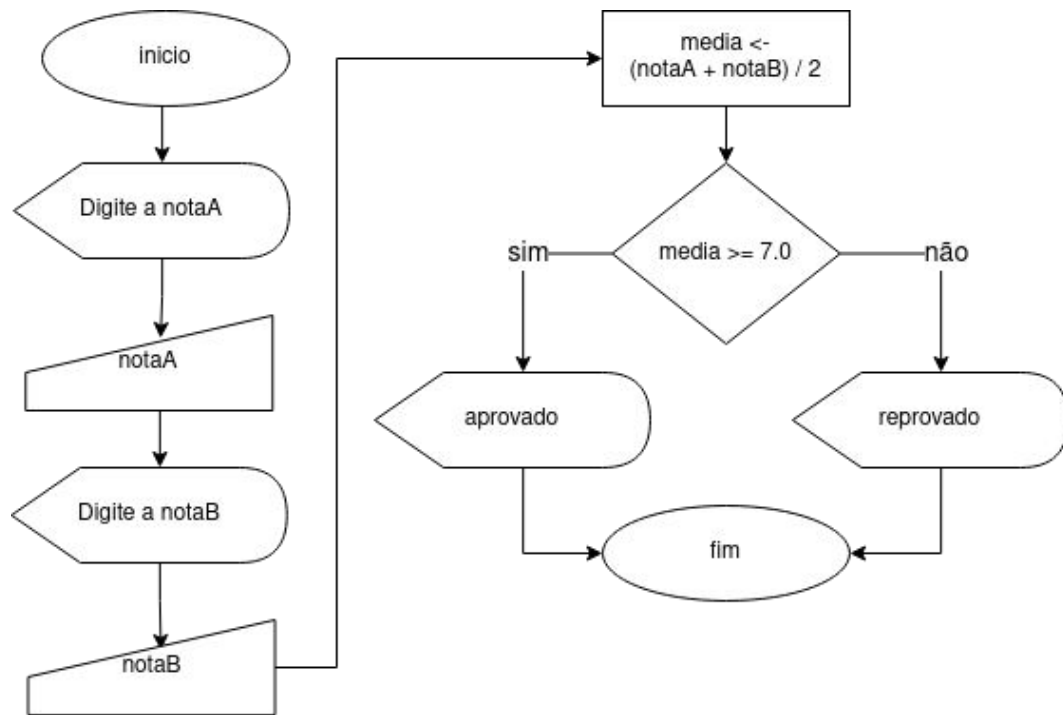
```
Digite a notaA: 5.6  
Digite a notaB: 3.5  
Reprovado
```

//Linguagem C

```
#include <stdio.h>  
  
int main() {  
    float notaA, notaB, media;  
    printf("Digite a notaA: ");  
    scanf("%f", &notaA);  
    printf("Digite a notaB: ");  
    scanf("%f", &notaB);  
  
    media = (notaA + notaB) / 2;  
    if (media >= 6.0) {  
        printf("Aprovado");  
    }  
    else {  
        printf("Reprovado");  
    }  
    return 0;  
}
```

# Estrutura condicional composta

- Estrutura condicional **IF**



# Estruturas condicionais aninhadas

- Muitas vezes será necessário tomar uma nova decisão dentro de um fluxo condicional; ou
- Pode ser que tenhamos mais de duas opções de fluxo de execução;
- Em ambos os casos, podemos utilizar estruturas condicionais aninhadas, que representam uma estrutura condicional inserida dentro de outra.

# Estrutura condicional aninhadas

```
Digite a notaA: 3.7
Digite a notaB: 4.1
Reprovado
```

```
float notaA, notaB, media;
```

```
escreva("Digite a notaA: ")
leia(notaA)
escreva("Digite a notaB: ")
leia(notaB)
media = (notaA + notaB) / 2
```

```
se (media >= 6.0) {
    escreva("Aprovado")
}
senao se (media >= 4.0) {
    escreva("Em recuperacao")
}
senao {
    escreva("Reprovado")
}
```

```
}
```

```
#include <stdio.h>
```

```
int main() {
    float notaA, notaB, media;
```

```
    printf("Digite a notaA: ");
    scanf("%f", &notaA);
    printf("Digite a notaB: ");
    scanf("%f", &notaB);
```

```
media = (notaA + notaB) / 2;

    se (media >= 6.0) {
        escreva("Aprovado");
    }
    senao se (media >= 4.0) {
        escreva("Em recuperacao");
    }
    senao {
        escreva("Reprovado");
    }
}
```

```
else if (media >= 4.0) {
    printf("Em recuperacao");
}
else {
    printf("Reprovado");
}
```

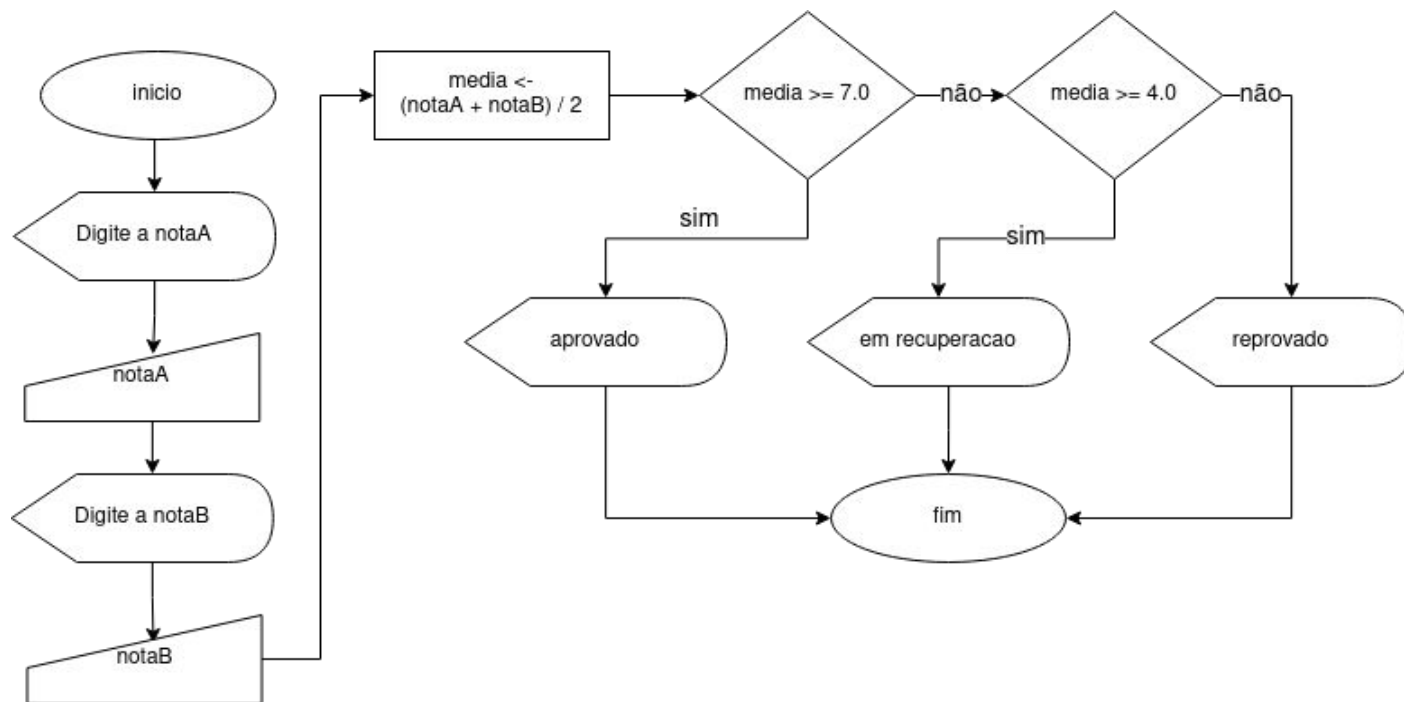
```
return 0;
```

```
}
```

```
Digite a notaA: 8.7
Digite a notaB: 6.8
Aprovado
```

# Estruturas Condicionais Aninhadas

- Estrutura condicional **IF - ELSEIF - ELSE**





# Condições Compostas em Estruturas Condicionais

- É possível compor expressões lógicas (condições) utilizando operadores relacionais:
  - `==, !=, <, >, <=, >=`
- Também é possível compor expressões lógicas utilizando operadores lógicos:
  - `nao(!), e(&&) e ou(||)`
- Então a expressão lógica que será avaliada também pode ser uma expressão lógica composta.

# Condições compostas em estruturas condicionais

```
programa {  
    funcao inicio() {  
  
        real freq, media  
  
        escreva("Digite a media: 8.7")  
        leia(media)  
        escreva("Digite a frequencia: 94.0")  
        leia(freq)  
  
        se ((media >= 7.0) e (freq >= 75.0)) {  
            escreva("Aprovado")  
        }  
        senao se ((media >= 4.0) e (freq >= 75.0)) {  
            escreva("Digite a media: 9.25")  
            leia(media)  
            escreva("Digite a frequencia: 87.5")  
            leia(freq)  
            se ((media >= 7.0) e (freq >= 75.0)) {  
                escreva("Aprovado")  
            }  
            senao {  
                escreva("Reprovado")  
            }  
        }  
    }  
}
```

```
#include <stdio.h>  
  
int main() {  
  
    float media, freq;  
  
    printf("Digite a media: 4.8")  
    scanf("%f", &media)  
    printf("Digite a frequencia: 95.0")  
    scanf("%f", &freq)  
    printf("Em recuperacao? ")  
    scanf("%f", &freq)  
  
    if ((media >= 7.0) e (freq >= 75.0)) {  
        printf("Aprovado");  
    }  
    else if ((media >= 4.0) e (freq >= 75.0)) {  
        printf("Digite a media: 3.8")  
        scanf("%f", &media)  
        printf("Digite a frequencia: 55.0")  
        scanf("%f", &freq)  
        if ((media >= 7.0) e (freq >= 75.0)) {  
            printf("Aprovado");  
        }  
        else {  
            printf("Reprovado");  
        }  
    }  
  
    return 0;  
}
```

# Outro jeito de aninhar condições

```
#include <stdio.h>

int main() {

    float freq, media;

    printf("Digite a media: ");
    scanf("%f", &media);

    printf("Digite a frequencia: ");
    scanf("%f", &freq);
```

→ *(segue no quadro ao lado)*

```
    if (freq >= 75.0) {

        if (media >= 6.0) {
            printf("Aprovado");
        } else if (media >= 4.0) {
            printf("Em recuperacao");
        } else {
            printf("Reprovado por media");
        }

    } else {
        printf("Reprovado por faltas");
    }

    return 0;
}
```

# Outro jeito de aninhar condições

```
programa {  
  funcao inicio() {  
  
    real freq, media  
  
    escreva("Digite a media: ")  
    leia(media)  
  
    escreva("Digite a frequencia: ")  
    leia(freq)
```

→ *(segue no quadro ao lado)*

```
    se (freq >= 75.0) {  
  
        se (media >= 6.0) {  
            escreva("Aprovado")  
        } senao se (media >= 4.0) {  
            escreva("Em recuperacao")  
        } senao {  
            escreva("Reprovado por media")  
        }  
  
    } senao {  
        escreva("Reprovado por faltas")  
    }  
  
}
```

# Estruturas condicionais – Síntese

ATENÇÃO para o uso de `{}`. O **else** é opcional e não tem condição.

```
if (<condição>) {  
    <bloco de comandos>  
}
```

```
if (<condição>) {  
    <bloco de comandos>  
} else {  
    <bloco de comandos>  
}
```

```
if (<condição1>) {  
    <bloco de comandos>  
} else if (<condição2>) {  
    <bloco de comandos>  
}
```

```
if (<condição1>) {  
    <bloco de comandos>  
} else if (<condição2>) {  
    <bloco de comandos>  
} else if (<condição3>) {  
    <bloco de comandos>  
} else {  
    <bloco de comandos>  
}
```

# Estruturas condicionais – Síntese

ATENÇÃO para o uso de **{}**. O **senao** é opcional e não tem condição.

```
se (<condição>) {  
    <bloco de comandos>  
}
```

```
se (<condição>) {  
    <bloco de comandos>  
} senao {  
    <bloco de comandos>  
}
```

```
se (<condição1>) {  
    <bloco de comandos>  
} senao se (<condição2>) {  
    <bloco de comandos>  
}
```

```
se (<condição1>) {  
    <bloco de comandos>  
} senao se (<condição2>) {  
    <bloco de comandos>  
} senao se (<condição3>) {  
    <bloco de comandos>  
} senao {  
    <bloco de comandos>  
}
```

# Próxima aula

- Retomada dos principais pontos desta aula;
- Estruturas condicionais simples, compostas e aninhadas (SE, ESCOLHA);
- Continuidade da introdução à Linguagem C.

# Dúvidas





# Obrigado!

Prof. Rhafael Freitas da Costa