



# Técnicas de Programação

## Estruturas de Seleção

Fábio Duncan de Souza

Instituto Federal Fluminense

- 1 Conceitos Básicos
- 2 Operadores Relacionais
- 3 Operadores Lógicos
- 4 Estrutura de Seleção Simples
- 5 Estrutura de Seleção Composta
- 6 Estruturas de Seleção Aninhadas
- 7 Estrutura de Seleção Múltipla



# Conceitos Básicos

- As estruturas de seleção simples permitem que um programa tome decisões com base na veracidade ou falsidade de alguma instrução ou fato chamado condição
  - Se a condição for atendida (condição verdadeira, ou true), a instrução no corpo da estrutura é executada;
  - Se a condição não for atendida (condição falsa, ou false), a instrução do corpo da estrutura não é executada;
- Sendo executada ou não a instrução do corpo, depois da estrutura de seleção ser concluída, a execução continua com a instrução após a estrutura.
- As condições nas estruturas de seleção são construídas usando operadores de igualdade, relacionais e lógicos.



# Operadores Relacionais

# Operadores Relacionais

- Os operadores relacionais possuem o mesmo nível de precedência e são associados da esquerda para a direita.
- Os operadores de igualdade possuem nível de precedência menor do que o dos operadores relacionais e também são associados da esquerda para a direita.

Operador	Em algoritmo	Na linguagem C
Maior que	>	>
Menor que	<	<
Maior ou igual	>=	>=
Menor ou igual	<=	<=
Igual	=	==
Diferente	≠ ou <>	!=

Obs: Cuidado para não confundir os operadores de igualdade (==) e de atribuição (=), na linguagem C.



# Operadores Lógicos

# Operadores Lógicos

Técnicas de  
Programação

Fábio Duncan

Conceitos  
Básicos

Operadores  
Relacionais

**Operadores  
Lógicos**

Estrutura de  
Seleção  
Simples

Estrutura de  
Seleção  
Composta

Estruturas de  
Seleção  
Aninhadas

Estrutura de  
Seleção  
Múltipla

- Os operadores lógicos servem para combinar resultados de expressões, retornando se o resultado final é verdadeiro ou falso;
- Viabilizam o teste de várias condições durante o processo de tomada de decisão.

Operador	Em algoritmo	Na linguagem C
E	AND	&&
OU	OR	
NÃO	NOT	!



# Operadores Lógicos

Técnicas de  
Programação

Fábio Duncan

Conceitos  
Básicos

Operadores  
Relacionais

**Operadores  
Lógicos**

Estrutura de  
Seleção  
Simples

Estrutura de  
Seleção  
Composta

Estruturas de  
Seleção  
Aninhadas

Estrutura de  
Seleção  
Múltipla

- Uma expressão com o operador **E** é verdadeira se todas as condições forem verdadeiras.
- Uma expressão com o operador **OU** é verdadeira se pelo menos uma condição for verdadeira.
- Uma expressão com o operador **NÃO** inverte o valor da expressão ou condição, se verdadeira inverte para falsa e vice-versa.



# Estrutura de Seleção Simples

# Estrutura de Seleção Simples

Técnicas de  
Programação

Fábio Duncan

Conceitos  
Básicos

Operadores  
Relacionais

Operadores  
Lógicos

Estrutura de  
Seleção  
Simples

Estrutura de  
Seleção  
Composta

Estruturas de  
Seleção  
Aninhadas

Estrutura de  
Seleção  
Múltipla

- Estrutura de Seleção Simples em Pseudocódigo
  - Implementada com o comando **SE...ENTÃO**;
  - A estrutura normalmente vem acompanhada de comandos, ou seja, se determinada condição for satisfeita pelo comando **SE...ENTÃO** então execute determinados comandos.

```
1 SE <condição> ENTÃO
2     <Comando1>
3     <Comando2>
4     <ComandoN>
5 FIM SE
```

# Estrutura de Seleção Simples

## Algoritmo em Pseudocódigo

Técnicas de  
Programação

Fábio Duncan

Conceitos  
Básicos

Operadores  
Relacionais

Operadores  
Lógicos

Estrutura de  
Seleção  
Simples

Estrutura de  
Seleção  
Composta

Estruturas de  
Seleção  
Aninhadas

Estrutura de  
Seleção  
Múltipla

- Fazer um programa para calcular a média de um aluno dadas as entradas das notas da prova 1 e da prova 2 pelo teclado. O programa deverá apresentar a mensagem "Parabéns" caso o aluno tenha sido aprovado com nota maior ou igual a 6.

```
1  Início
2      Declare
3          real notaP1, notaP2, mediaNotas
4
5      escreva("Entre com a nota da P1")
6      leia(notaP1)
7      escreva("Entre com a nota da P2")
8      leia(notaP2)
9      media ← (notaP1 + notaP2) / 2
10     se (media >= 6) então
11         escreva("Parabéns")
12     fim se
13 Fim Algoritmo
```

- Estrutura de Seleção Simples na Linguagem C
  - Implementada com o comando **if**;
  - Por ser um comando de bloco, faz-se necessário o uso de chaves delimitadoras
    - Obs: O uso das chaves para blocos de código com um único comando é facultativo, porém recomendo que, durante o aprendizado, esta exceção seja ignorada.

```
1  if (<condição>)
2  {
3      <Comando1>
4      <Comando2>
5      <ComandoN>
6  }
```



# Estrutura de Seleção Simples

## Linguagem C

Técnicas de  
Programação

Fábio Duncan

Conceitos  
Básicos

Operadores  
Relacionais

Operadores  
Lógicos

Estrutura de  
Seleção  
Simples

Estrutura de  
Seleção  
Composta

Estruturas de  
Seleção  
Aninhadas

Estrutura de  
Seleção  
Múltipla

- Fazer um programa para calcular a média de um aluno dadas as entradas das notas da prova 1 e da prova 2 pelo teclado. O programa deverá apresentar a mensagem "Parabéns" caso o aluno tenha sido aprovado com nota maior ou igual a 6.

```
1  #include <stdio.h>
2
3  int main()
4  {
5      float notaP1, notaP2, media;
6      printf("Entre com a nota da P1:");
7      scanf("%f", &notaP1);
8      printf("Entre com a nota da P2:");
9      scanf("%f", &notaP2);
10     media = (notaP1 + notaP2) / 2.0;
11     if (media >= 6)
12     {
13         printf("Parabéns!");
14     }
15     return 0;
16 }
```



## Operadores Relacionais

## Operadores Lógicos

## Estrutura de Seleção Simples

## Estruturas de Seleção Aninhadas

## Estrutura de Seleção Múltipla

# Estrutura de Seleção Composta



- A estrutura de seleção composta funciona como a estrutura de seleção simples, com a diferença de que, na simples somente são executados comandos caso a condição seja verdadeira, já na composta pelo menos um comando será executado independente da condição;
- Caso a condição seja “verdadeira” os comandos da condição “verdadeira” serão executados, caso contrário os comandos da condição “falsa” serão executados.





# Estrutura de Seleção Composta

## Algoritmo em Pseudocódigo

- Fazer um programa para calcular a média de um aluno dadas as entradas das notas da prova 1 e da prova 2 pelo teclado. O programa deverá apresentar a mensagem "Aprovado" caso o aluno tenha sido aprovado com nota maior ou igual a 6 e a mensagem "Reprovado" caso contrário.

```
1  Início
2      Declare
3          real notaP1, notaP2, mediaNotas
4
5      escreva("Entre com a nota da P1")
6      leia(notaP1)
7      escreva("Entre com a nota da P2")
8      leia(notaP2)
9      media ← (notaP1 + notaP2) / 2
10     se (media >= 6) então
11         escreva("Aprovado")
12     senão
13         escreva("Reprovado")
14     fim se
15 Fim Algoritmo
```



# Estrutura de Seleção Composta

- Estrutura de Seleção Simples na Linguagem C
  - Implementada com o comando **if...else**;
  - Por ser um comando de bloco, faz-se necessário o uso de chaves delimitadoras em ambos os blocos.

```
1  if (<condição>)
2  {
3      <Comando1>
4      <Comando2>
5      <ComandoN>
6  }
7  else
8  {
9      <Comando1>
10     <Comando2>
11     <ComandoN>
12 }
```



# Estrutura de Seleção Composta

## Linguagem C

- Fazer um programa para calcular a média de um aluno dadas as entradas das notas da prova 1 e da prova 2 pelo teclado. O programa deverá apresentar a mensagem "Aprovado" caso o aluno tenha sido aprovado com nota maior ou igual a 6 e a mensagem "Reprovado" caso contrário.

```
1 #include <stdio.h>
2
3 int main()
4 {
5     float notaP1, notaP2, media;
6     printf("Entre com a nota da P1:");
7     scanf("%f", &notaP1);
8     printf("Entre com a nota da P2:");
9     scanf("%f", &notaP2);
10    media = (notaP1 + notaP2) / 2.0;
11    if (media >= 6){
12        printf("Aluno Aprovado");
13    }
14    else {
15        printf("Aluno Reprovado");
16    }
17    return 0;
18 }
```



**INSTITUTO  
FEDERAL**  
Fluminense

Técnicas de Programação

## Conceitos Básicos

## Operadores Relacionais

## Operadores Lógicos

## Estrutura de Seleção Simples

## Estrutura de Seleção Composta

## Estrutura de Seleção Múltipla

## Estruturas de Seleção Aninhadas



# Estruturas de Seleção Aninhadas

- Para se alcançar uma determinada lógica de decisão, é comum que estruturas de seleção sejam incluídas nas condições falsas de outras estruturas de seleção, dando início a um encadeamento de estruturas.

```
1 SE <condição> ENTÃO
2     <Comando1>
3     <Comando2>
4     <ComandoN>
5 SENÃO
6     SE <condição> ENTÃO
7         <Comando1>
8         <Comando2>
9         <ComandoN>
10    SENÃO
11        <Comando1>
12        <Comando2>
13        <ComandoN>
14    FIM SE
15 FIM SE
```







### ● Solução 1 - Pseudocódigo

```
1  Início
2      Declare
3          real notaP1, notaP2, media
4      leia(notaP1)
5      leia(notaP2)
6      media = (notaP1 + notaP2) / 2
7      se media >= 9 então
8          escreva("Conceito A!")
9      senão
10         se (media >= 7.5) E (media <= 8.9) então
11             escreva("Conceito B!")
12         senão
13             se (media >= 6) E (media <= 7.4) então
14                 escreva("Conceito C!")
15             senão
16                 se (media >= 4) E (media <= 5.9) então
17                     escreva("Conceito D!")
18                 senão
19                     escreva("Conceito E!")
20             fim se
21         fim se
22     fim se
23 fim se
24 Fim Algoritmo
```



### • Solução 2 - Pseudocódigo

```
1  Início
2      Declare
3          real notaP1, notaP2, media
4      leia(notaP1)
5      leia(notaP2)
6      media = (notaP1 + notaP2) / 2
7      se media >= 9 então
8          escreva("Conceito A!")
9      senão
10         se media >= 7.5 então
11             escreva("Conceito B!")
12         senão
13             se media >= 6 então
14                 escreva("Conceito C!")
15             senão
16                 se media >= 4 então
17                     escreva("Conceito D!")
18                 senão
19                     escreva("Conceito E!")
20             fim se
21         fim se
22     fim se
23 fim se
24 Fim Algoritmo
```



### ● Solução 1 - Linguagem C - Parte 2

```
1  if (media >= 9) {  
2      printf("Conceito A!\n");  
3  }  
4  else {  
5      if ((media >= 7.5) && (media <= 8.9)) {  
6          printf("Conceito B!\n");  
7      }  
8      else {  
9          if ((media >= 6) && (media <= 7.4)) {  
10             printf("Conceito C!\n");  
11         }  
12         else {  
13             if ((media >= 4) && (media <= 5.9)) {  
14                 printf("Conceito D!\n");  
15             }  
16             else {  
17                 printf("Conceito E!\n");  
18             }  
19         }  
20     }  
21 }  
22 return 0;  
23 }
```

## Estrutura de Seleção Múltipla

# Estrutura de Seleção Múltipla

Técnicas de  
Programação

Fábio Duncan

Conceitos  
Básicos

Operadores  
Relacionais

Operadores  
Lógicos

Estrutura de  
Seleção  
Simples

Estrutura de  
Seleção  
Composta

Estruturas de  
Seleção  
Aninhadas

**Estrutura de  
Seleção  
Múltipla**

- O conteúdo de uma variável é comparado com um valor constante, e caso a comparação seja verdadeira, um determinado comando é executado.
- Permite a escolha de um grupo de sequência de operações, dependendo do valor de uma variável seletora;
- O valor seletor é comparado com cada valor de uma sequência de valores, e se algum valor comparado for igual ao seletor, a sequência de operações relacionadas a ele será executada;

- Caso nenhuma igualdade seja obtida, nenhuma operação é executada por ele, passando então para o comando seguinte a estrutura;
- Pode ser utilizada uma espécie de **senão**, que será executado caso nenhuma das demais opções o sejam;
- É comumente utilizada em menus;
- Não são aceitas expressões relacionais e lógicas neste tipo de estrutura.

# Estrutura de Seleção Múltipla

Técnicas de  
Programação

Fábio Duncan

Conceitos  
Básicos

Operadores  
Relacionais

Operadores  
Lógicos

Estrutura de  
Seleção  
Simples

Estrutura de  
Seleção  
Composta

Estruturas de  
Seleção  
Aninhadas

Estrutura de  
Seleção  
Múltipla

- Estrutura de Seleção Múltipla Pseudocódigo
  - Implementada com o comando **ESCOLHA...CASO**;

```
1 ESCOLHA <variavel>
2     CASO <Valor 1>
3         <sequência 1>
4     CASO <Valor 2>
5         <sequência 2>
6         .....
7     CASO <Valor n>
8         <sequência n>
9     OUTRO CASO
10        <sequência padrão>
11        <para os valores diferentes de todos os anteriores>
12 FIM ESCOLHA
```





# Estrutura de Seleção Múltipla

## Exemplo de Algoritmo em Pseudocódigo

- Prepare um algoritmo para entrar pelo teclado com um número correspondente a um dia da semana. O algoritmo deverá imprimir o nome do dia da semana por extenso.

```
1  Início
2      inteiro numeroDoDiaDaSemana
3      leia(numeroDoDiaDaSemana)
4      escolha(numeroDoDiaDaSemana)
5          caso 1
6              escreva("Domingo")
7          caso 2
8              escreva("Segunda-Feira")
9          caso 3
10             escreva("Terça-Feira")
11          caso 4
12             escreva("Quarta-feira")
13          caso 5
14             escreva("Quinta-Feira")
15          caso 6
16             escreva("Sexta-Feira")
17          caso 7
18             escreva("Sábado")
19          outro caso
20             escreva("Este número de dia da semana não é válido")
21      fim escolha
22  Fim algoritmo
```

# Estrutura de Seleção Múltipla

- Estrutura de Seleção Múltipla na Linguagem C
  - Implementada com o comando **switch...case**;
  - Suporta variáveis do tipo inteiro e caractere.

```
1 switch (variável)
2 {
3     case constante1:
4         Instruções;
5     break;
6     case constante2:
7         Instruções;
8     break;
9     default:
10        Instruções;
11 }
```



# Estrutura de Seleção Múltipla

## Exemplo de Programa na Linguagem C

- Prepare um programa para entrar com um número correspondente a um dia da semana. O algoritmo deverá imprimir o nome do dia da semana por extenso.

```
1 #include <stdio.h>
2 int main() {
3     int numeroDoDiaDaSemana;
4     printf("Entre com o numero do dia da semana [1 a 7]:");
5     scanf("%d", &numeroDoDiaDaSemana);
6     switch (numeroDoDiaDaSemana) {
7         case 1: printf("\nDomingo!\n");
8                 break;
9         case 2: printf("\nSegunda-Feira!\n");
10                break;
11        case 3: printf("\nTerca-Feira!\n");
12                break;
13        case 4: printf("\nQuarta-feira!\n");
14                break;
15        case 5: printf("\nQuinta-Feira!\n");
16                break;
17        case 6: printf("\nSexta-Feira!\n");
18                break;
19        case 7: printf("\nSabado!\n");
20                break;
21        default: printf("\nEste número de dia da semana não é válido!\n");
22    }
23    return 0;
24 }
```

# Referências Bibliográficas

Técnicas de  
Programação

Fábio Duncan

Conceitos  
Básicos

Operadores  
Relacionais

Operadores  
Lógicos

Estrutura de  
Seleção  
Simples

Estrutura de  
Seleção  
Composta

Estruturas de  
Seleção  
Aninhadas

Estrutura de  
Seleção  
Múltipla



Harvey M Deitel and Paul J Deitel.  
*Como programar em C.*  
LTC, 1999.



Renato Cardoso Mesquita.  
*Curso de linguagem C.*  
Universidade Federal de Minas Gerais, 1998.



Wikimedia.  
*Linguagem de programação.*  
Wikipédia: a enciclopédia livre, 2020.