

Estruturas de Controle - estrutura condicional -

Algoritmos I
2014/1

Profa: Daniela Scherer dos Santos
daniela.santos37@ulbra.edu.br
www.danielascherer.com.br

Estruturas de Controle

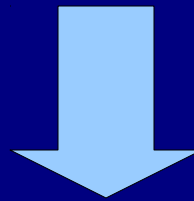
- ♦ A estrutura de controle de um algoritmo indica o fluxo de execução que deverá ser seguido pelo algoritmo para que ele processe a saída desejada;
- ♦ Estruturas básicas de controle:
 - ♦ sequencial;
 - ♦ **de seleção (condicional);**
 - ♦ de repetição.

Exercício exemplo:

1) Escreva um algoritmo que calcule a média aritmética entre quatro notas (N1, N2, N3, N4), apresente o resultado obtido e em seguida a mensagem “ALUNO APROVADO” se a média for superior ou igual a seis.

Exercício exemplo:

1) Escreva um algoritmo que calcule a média aritmética entre quatro notas (N1, N2, N3, N4), apresente o resultado obtido e em seguida a mensagem “ALUNO APROVADO” se a média for superior ou igual a seis.



para resolver esse problema é necessário o entendimento de um novo comando → COMANDO DE SELEÇÃO

Estrutura de Seleção

- uma estrutura de seleção permite a escolha de um grupo de ações (bloco) a ser executado quando determinadas condições, representadas por expressões lógicas ou relacionais, são ou não satisfeitas.

Estrutura de Seleção

- ♦ Seleção simples;
- ♦ Seleção composta;
- ♦ Seleção encadeada;
- ♦ Seleção de múltipla escolha.

Estrutura de Seleção

- ♦ Seleção simples;
- ♦ Seleção composta;
- ♦ Seleção encadeada;
- ♦ Seleção de múltipla escolha.

Seleção Simples

- ◆ Quando precisamos testar uma certa condição antes de executar uma ação, usamos uma *seleção simples*;
- ◆ SINTAXE:

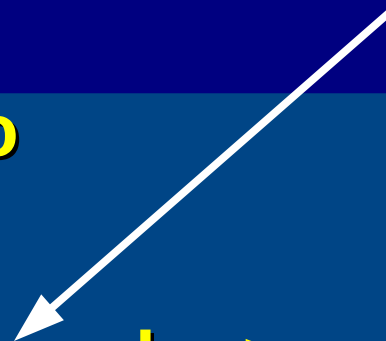
se (condição) então

inicio

<bloco de comandos>;

fim

ações



Seleção Simples

condição → expressão lógica ou relacional que, quando testada, pode gerar um resultado falso ou verdadeiro. Se a condição for verdadeira, a sequência de ações descritas no bloco (entre início e fim) serão executadas. Caso contrário (condição for falsa), encerra-se a seleção sem executar nenhum comando.

♦ SINTAXE:

se (condição) então

início

<bloco de comandos>;

fim

Estrutura de Seleção

- ♦ Seleção simples;
- ♦ **Seleção composta;**
- ♦ Seleção encadeada;
- ♦ Seleção de múltipla escolha.

Seleção Composta

1) Escreva um algoritmo que calcule a média aritmética entre quatro notas (N1, N2, N3, N4), apresente o resultado obtido e em seguida a mensagem “ALUNO APROVADO” se a média for superior ou igual a seis, “ALUNO REPROVADO” se a média for inferior a seis.

Seleção Composta

- Quando tivermos situações em que duas alternativas dependem de uma mesma condição, uma da condição ser verdadeira e outra da condição ser falsa, usamos a estrutura de seleção composta.

Seleção Composta

- SINTAXE:

```
se (condição) então
    inicio
        comando 1;
        comando 2;
    fim
senão
    inicio
        comando 3;
        comando 4;
    fim
```

Se a condição for verdadeira, os comandos 1 e 2 serão executados. Caso contrário (condição for falsa), os comandos 3 e 4 serão executados.

Estrutura de Seleção

- ♦ Seleção simples;
- ♦ Seleção composta;
- ♦ Seleção encadeada;
- ♦ Seleção de múltipla escolha.

Seleção Encadeada

- Quando, devido à necessidade de processamento, agruparmos várias seleções, formaremos uma seleção encadeada.

Seleção Encadeada

- SINTAXE:

```
se (condição1) então
  inicio
    se (condição2) então
      inicio
        <sequência de comandosA>;
      fim
    fim
  fim
senão se (condição3) então
  inicio
    <sequência de comandosB>;
  fim
senão se (condição4) então
  inicio
    se (condição5) então
      inicio
        <sequência de comandosC>;
      fim
    senão
      inicio
        <sequência de comandosD>;
      fim
    fim
  fim
```


Seleção Encadeada

- SINTAXE:

“sequência de comandos A” somente será executada se a *condição1* e a *condição2* forem verdadeiras

```
se (condição1) então
    início
        se (condição2) então
            início
                <sequência de comandosA>;
            fim
        fim
    fim
senão se (condição3) então
    início
        <sequência de comandosB>;
    fim
senão se (condição4) então
    início
        se (condição5) então
            início
                <sequência de comandosC>;
            fim
        senão
            início
                <sequência de comandosD>;
            fim
        fim
    fim
fim
```

Seleção Encadeada

- SINTAXE:

“sequência de comandos B” somente será executada se a *condição1* for falsa e a *condição3* for verdadeira

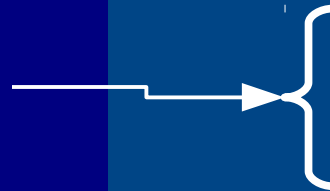
```
se (condição1) então
  inicio
    se (condição2) então
      inicio
        <sequência de comandosA>;
      fim
    fim
  fim
  senão se (condição3) então
    inicio
      <sequência de comandosB>;
    fim
  senão se (condição4) então
    inicio
      se (condição5) então
        inicio
          <sequência de comandosC>;
        fim
      senão
        inicio
          <sequência de comandosD>;
        fim
    fim
  fim
fim
```

Seleção Encadeada

- SINTAXE:

“sequência de comandos C” somente será executada se a *condição1* e a *condição3* forem falsas e as *condições* 4 e 5 forem verdadeiras

```
se (condição1) então
  inicio
    se (condição2) então
      inicio
        <sequência de comandosA>;
      fim
    fim
  fim
senão se (condição3) então
  inicio
    <sequência de comandosB>;
  fim
senão se (condição4) então
  inicio
    se (condição5) então
      inicio
        <sequência de comandosC>;
      fim
    senão
      inicio
        <sequência de comandosD>;
      fim
    fim
  fim
fim
```




Seleção Encadeada

- SINTAXE:

“sequência de comandos D” somente será executada se a *condição1* e a *condição3* forem falsas, a *condição4* for verdadeira e a *condição5* for falsa.

```
se (condição1) então
  inicio
    se (condição2) então
      inicio
        <sequência de comandosA>;
      fim
    fim
  fim
senão se (condição3) então
  inicio
    <sequência de comandosB>;
  fim
senão se (condição4) então
  inicio
    se (condição5) então
      inicio
        <sequência de comandosC>;
      fim
    senão
      inicio
        <sequência de comandosD>;
      fim
    fim
  fim
```



Estrutura de Seleção

- ♦ Seleção simples;
- ♦ Seleção composta;
- ♦ Seleção encadeada;
- ♦ Seleção de múltipla escolha.

Seleção de Múltipla Escolha

- ◆ Quando um conjunto de valores precisa ser testado para uma variável e ações diferentes são associadas a esses valores, estamos diante de uma seleção encadeada onde pode-se utilizar uma estrutura específica denominada seleção de múltipla escolha;
- ◆ Exemplos:
 - ◆ testar o código de um produto;
 - ◆ testar uma operação a ser executada (+, *, /, -);

Seleção de Múltipla Escolha

- ♦ SINTAXE:

escolha seletor

inicio

caso V1: <bloco de instruções A>;

caso V2: <bloco de instruções B>;

caso V3: <bloco de instruções C>;

caso Vn: <bloco de instruções D>;

caso contrário: <bloco de instruções E>;

fim

onde:

seletor é uma variável cujo conteúdo precisa ser testado (não pode ser do tipo real);

V1..Vn são os valores esperados para o seletor.

Seleção de Múltipla Escolha

- ◆ Exemplo: faça um algoritmo que leia dois números reais e um caracter. Conforme o valor lido para o caracter, o algoritmo deverá calcular e informar:
 - ◆ se caracter for +, deverá somar os dois valores;
 - ◆ se caracter for -, deverá subtrair os dois valores;
 - ◆ se caracter for *, deverá multiplicar os dois valores;
 - ◆ se o caracter for /, deverá dividir os dois valores.

Seleção de Múltipla Escolha

```
algoritmo calculadora;  
var  
    real num1, num2;  
    caracter operador;  
inicio  
    leia(num1,num2);  
    leia(operador);  
    escolha operador  
    inicio  
        caso '+':  
            escreva ("A soma é:", num1+num2);  
  
        caso '-':  
            escreva ("A diferença é:", num1-num2);  
  
        caso '*':  
            escreva ("A multiplicação é:", num1*num2);  
  
        caso '/':  
            escreva ("A divisão é:", num1/num2);  
  
        caso contrário:  
            escreva ("Operador inválido");  
  
    fim  
fim
```

* o conteúdo da variável *operador* é testado entre ' ' por se tratar de um caracter apenas

FORBELLONE, A. L. V.; EBERSPÄCHER, H. F.. Lógica de Programação: a construção de algoritmos e estruturas de dados. São Paulo: Makron Books, 2000.