

Algoritmos e Lógica de Programação

Construção de Algoritmos

6

CEFET-AL
CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA

Estruturas de Decisão
Teste de Corretude do Algoritmo

Reinaldo Gomes
reinaldo@cefet-al.br

Estruturas de Decisão

- Com o conhecimento adquirido até aqui, ainda não é possível tomar decisões!
- Como escrever algoritmos que selecionem o que fazer?
 - Usar Estruturas de Decisão!

Estruturas de Decisão

- São estrutura que, dada a avaliação de uma expressão lógica (condição), permitem a escolha de um fluxo de instruções a ser executado
 - No máximo só podem existir dois fluxos de instruções: um se a condição for V e outro se a condição for F
- Os tipos de estruturas de decisão são 3:
 - Decisão Simples
 - Decisão Composta
 - Decisão Múltipla

Estruturas de Decisão Simples

- Na decisão simples, uma única expressão lógica é avaliada
- Sintaxes:


```
Se <expressão lógica> Entao
    <comando único>;

Se <expressão lógica> Entao
    Inicio
        <comando 1>;
        ...
        <comando N>;
    Fim;
```

Sequência

Estruturas de Decisão Simples

- Semântica: Se a avaliação da condição for V, um conjunto de comandos do bloco verdade é executado e prossegue-se o fluxo. Caso a avaliação da condição for F, salta-se para o primeiro comando após o fim da estrutura de decisão simples

Estruturas de Decisão Simples

Algoritmo ExemploDecisãoSimples

```
Inteiro : x;
Inicio
    Escreva("Digite um valor");
    Leia (x);
    Se (x > 0) Então
        Escreva ("X > 0");
Fim
```

Estruturas de Decisão Composta

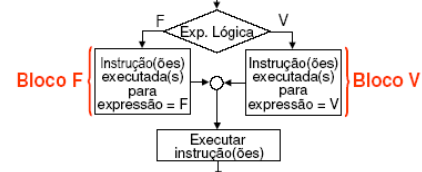
- Além de executar se a condição for V a decisão composta também executa se a condição for F.
- Sintaxes:

Se <expressão lógica> Entao <comando único>; Senao <comando único>	Se <expressão lógica> Entao Inicio <comando 1> ... <comando N> Fim; Senao Sequência;
---	---

7

Estruturas de Decisão Composta

- Semântica: Se a avaliação da condição for V, um conjunto de comandos do bloco verdade é executado e prossegue-se o fluxo. Caso a avaliação da condição for F, um conjunto de comandos do bloco falsidade é executado e prossegue-se o fluxo.



8

Estruturas de Decisão Composta

Algoritmo ExemploDecisãoComposta

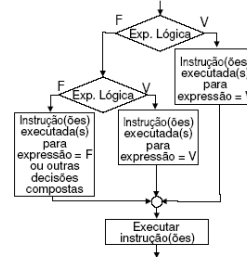
Inteiro: x;
 Início
 Escreva("Digite um valor");
 Leia (x);
 Se (x > 0) Então
 Escreva ("X > 0");
 Senão
 Escreva ("X <= 0");
 Fim.



9

Estruturas de Decisão Composta

- O encadeamento de decisões composta é utilizado para aninhar as estruturas Se Então Senão



10

Estruturas de Decisão Composta

- Questão de Implementação I
- Esta abordagem só é indicada quando há a necessidade de se testar condições dentro de condições. Nestes casos, normalmente, torna o algoritmo mais rápido, pois menos testes podem ser efetuados!

11

Estruturas de Decisão Composta

Algoritmo ExemploDecisãoCompostaAninhada

Inteiro: x;
 Início
 Escreva("Digite um valor");
 Leia (x);
 Se (x > 0) Então
 Escreva ("X > 0");
 Senão
 Se (x < 0) Então
 Escreva ("X < 0");
 Senão
 Escreva ("X = 0");
 Fim.



12

Estruturas de Decisão Composta

Se $(x > 0)$ Então
Escreva ("X > 0");

Se $(x < 0)$ Então
Escreva ("X < 0");

Se $(x = 0)$ Então
Escreva ("X = 0");

Isso vai funcionar,
mas é uma péssima prática!

Se um número é >0 , então ele não pode ser $=0$ ou <0 .
Por isso, o exemplo acima faz testes desnecessários!
Note que todas as condições SEMPRE são testadas!
Isso torna a execução do algoritmo mais lenta!

Estruturas de Decisão Composta

- Questão de Implementação II
- Se existem condições mutuamente exclusivas (somente uma, e apenas uma, condição pode ser V), então sempre deve-se usar DECISÃO COMPOSTA para se obter algoritmos mais rápidos!

14

Estruturas de Decisão Composta

Se $(x \text{ MOD } 2 = 0)$ Então
Se $(x > 0)$ Então
Se $(x/2 = 0)$ Então
Escreva ("X é par, positivo e múltiplo de 2");

O encadeamento acima é o mesmo que:
SE $(x \text{ MOD } 2 = 0)$ E $(x > 0)$ E $(x/2 = 0)$ Então
Escreva ("X é par, positivo e múltiplo de 2");

15

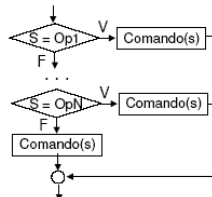
Estruturas de Decisão Múltipla

- A decisão múltipla é uma especialização da estrutura de decisão composta aninhada, que permite a execução de opções mutuamente exclusivas
- Sintaxe:
Escolha <Seletor>
<Opção1> : <instrução(ões)>;
...
<OpçãoN> : <instrução(ões)>;
[Senão <comando(s)> ;] opcional!
Fim Escolha;

16

Estruturas de Decisão Múltipla

- Semântica: realiza a comparação de igualdade para identificar qual opção tem o mesmo valor do seletor. Caso todas as condições sejam F, executa-se o senão (se este existir) ou saísse da estrutura sem executar nada



17

Estruturas de Decisão Múltipla

```
Algoritmo ExemploDecisãoMúltipla
Caracter: x;
Início
  Escreva("Digite um caracter");
  Leia (x);
  Escolha (x)
    "A","E","I","O","U" : Escreva ("Vogal");
    "-", "A", "v" : Início
      Escreva("Operador");
      Escreva(" Lógico");
    Fim;
  Senão Escreva ("Caracter não reconhecido");
Fim Escolha;
Fim.
```

18

Teste de Corretude do Algoritmo

- Após desenvolver um algoritmo ele deverá ser testado
- Este teste é chamado de TESTE DE MESA, que significa, seguir as instruções do algoritmo para verificar se a solução está correta

19

Teste de Corretude do Algoritmo

Algoritmo ExemploTesteDeMesa

Real: nota1, nota2, média;

Literal: situação;

Início

Ler (nota1, nota2);

média ← (nota1+nota2)/2;

Se (média >= 7) então

 situação ← "Aprovado";

Senão

 situação ← "Reprovado";

Escreva (Situação);

Fim.

Tabela de variáveis			
Nota1	Nota2	Média	Situação
7,0	8,5	7,75	Aprovado
6,5	7,0	6,75	Reprovado
8,0	6,0	7,0	Aprovado
5,0	7,5	6,25	Reprovado

20

Construção de Algoritmos

6



Estruturas de Decisão
Teste de Corretude do Algoritmo