

Estruturas de Seleção



PROFESSOR: DIEGO RICARDO KROHL
`diego.krohl@ifc-videira.edu.br`

Estruturas de seleção - Introdução



- Existem problemas que podem ter mais de um caminho para serem resolvidos em uma seleção correta. Muitas vezes também podem existir restrições nas soluções. Portanto, para executar o algoritmo, em dado momento é preciso tomar a decisão do caminho que deverá ser seguido para chegar à solução correta para o problema.

Estruturas de seleção - Introdução



- A decisão do caminho a ser seguido precisa estar fundamentada em alguma lógica. E é função de quem escreve o algoritmo dar condições para o sujeito que o executa fazer a escolha correta do caminho para se chegar à solução do problema.

Estruturas de seleção - Introdução



- Um exemplo simples de problema que pode ter metodologias diferentes em sua solução são os cálculos das raízes de equações polinomiais de 2º grau ($ax^2 + bx + c = 0$). Se o termo $b^2 - 4ac$ for maior ou igual a zero, as raízes serão reais. Caso contrário, serão complexas.

Estruturas de seleção - Introdução



- Não seria correto escrever um algoritmo que resolvesse somente a parte de raízes reais ou a parte complexa, pois a metodologia que vai ser usada somente será conhecida na execução do algoritmo.
- Logo, é preciso descrever a solução por ambas as metodologias, mas deixando claro para quem irá executá-la que será preciso fazer uma escolha entre os caminhos possíveis.

Estruturas de seleção - Aplicação



- Um problema compreende as seguintes situações:
 - Distribuição gratuita de cestas básicas.
 - Distribuição gratuita de cestas básicas para as famílias com quatro ou mais componentes;
 - Distribuição gratuita de ingressos para o teatro, sendo dois para pessoas do sexo feminino e um para pessoas do sexo masculino.

Estruturas de seleção - Aplicação

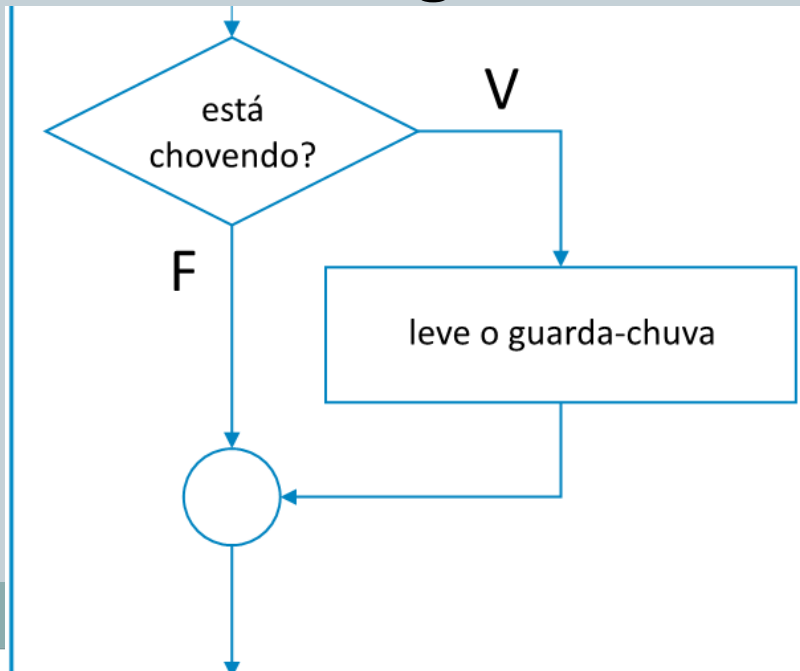


- Levando em consideração as restrições anteriores chegamos às seguintes conclusões:
 - Na primeira, todas as pessoas recebem a cesta básica, o que equivaleria a um comando sequencial;
 - Na segunda, só recebem as cestas básicas as famílias com pelo menos quatro integrantes;
 - Na terceira, dependendo do sexo, recebe-se um ou dois ingressos.

Comando SE



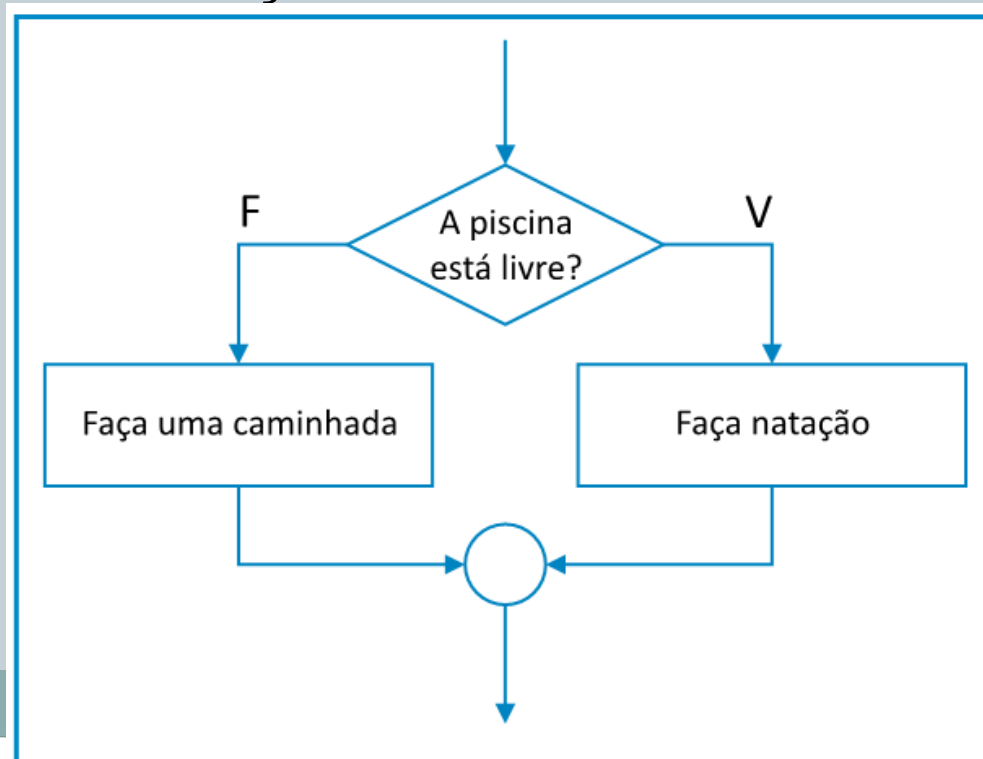
- É muito comum ver situações em programação, assim como na prática, em que se decide por um ou por outro procedimento, de acordo com certas condições. Por exemplo, ao dizer “SE estiver chovendo ENTÃO leve um guarda-chuva”.



Comando SE/ENTÃO



- Em alguns casos, especifica-se também o que deverá ser feito caso a condição não se cumpra, onde: “SE a piscina estiver livre ENTÃO faça natação, CASO CONTRÁRIO faça uma caminhada”.



Comando SE/ENTAO



- Essas situações são muito comuns em programação, razão pela qual será preciso recorrer aos comandos de seleção (ou de decisão);
- A condição Se-Então (-Senão) é uma estrutura de seleção comum em diversas linguagens de programação.

```
algoritmo "SE"  
var  
    varNum : inteiro  
inicio  
    se varNum < 10 entao  
        escreva ("Número menor que 10")  
    senão  
        escreva ("Número maior ou igual a 10")  
    fimse  
finalgoritmo
```

Ninhos de comandos **SE**



- Dentro de um comando de seleção (comando Se) é possível colocar qualquer tipo de comando.
- Assim, dentro de um comando Se, pode-se ter outros comandos de seleção.

```
algoritmo "MAXIMO"  
var  
    varNum1, varNum2 : in-  
teiro  
inicio  
    leia (varNum1)  
    leia (varNum2)  
    se varNum1 > varNum2  
entao  
        escreva ("Maior valor é  
o primeiro")  
    senão  
        escreva ("Maior valor é  
o segundo")  
    fimse  
finalgoritmo
```

Ninhos de comandos **SE**



```
algoritmo "MAXIMO"  
var  
    varNum1, varNum2 : inteiro  
inicio  
    leia (varNum1)  
    leia (varNum2)  
    se varNum1 > varNum2 entao  
        escreva ("Maior valor é o primeiro")  
    senão  
        se varNum1 < varNum2  
            escreva ("Maior valor é o segundo")  
        senão  
            escreva ("Os dois valores são iguais")  
        fimse  
    fimse  
fimalgoritmo
```

Comando ESCOLHA



- Quando um conjunto de valores discretos precisa ser testado e ações diferentes são associadas a esses valores, estamos diante de uma seleção com ninhos de comando Se. Como essa situação é bastante frequente na construção de algoritmos que dependem de alternativas, utilizaremos uma estrutura específica para esses casos: a seleção de múltipla escolha.
- O comando Escolha é uma alternativa para os comandos Se aninhados, deixando o algoritmo com uma estrutura melhor.

Comando **ESCOLHA**



- Utilizando SE / SENAO aninhados:

```
algoritmo "CONCEITO"
var
    varNota : inteiro
    varConceito : caractere
inicio
    leia (varNota)
    se varNota = 1 entao
        varConceito ← "Insuficiente"
    senão
        se varNota = 2
            varConceito ← "Regular"
        senão
            se varNota = 3
                varConceito ← "Bom"
            senão
                varConceito ← "Ótimo"
        fimse
    fimse
fimse
escreva (varConceito)
finalgoritmo
```

Comando **ESCOLHA**



- Utilizando ESCOLHA:

```
algoritmo "CONCEITO"  
var  
    varNota : inteiro  
    varConceito : caractere  
inicio  
    leia (varNota)  
    escolha varNota  
    caso 1  
        varConceito ← "Insuficiente"  
    caso 2  
        varConceito ← "Regular"  
    caso 3  
        varConceito ← "Bom"  
    outrocaso  
        varConceito ← "Ótimo"  
    fimescolha  
    escreva (varConceito)  
finalgoritmo
```

Comando **ESCOLHA**



- Exemplo:

```
algoritmo "CONCEITO"  
var  
    varNota : inteiro  
    varConceito : caractere  
inicio  
    leia (varNota)  
    escolha varNota  
    caso 1  
        varConceito ← "Insuficiente"  
    caso 2  
        varConceito ← "Regular"  
    caso 3  
        varConceito ← "Bom"  
    outrocaso  
        varConceito ← "Ótimo"  
    fimescolha  
    escreva (varConceito)  
finalgoritmo
```