

CONTROLE DE FLUXO DE EXECUÇÃO

Controle de fluxo de execução

- É a sequência em que as instruções são executadas num algoritmo.

Como podemos implementar o controle de fluxo

- **Instrução composta**

- É um conjunto de comandos simples como atribuição, entrada, saída ou algumas construções (estruturas).

- **Estrutura sequencial**

- Cada comando é executado somente após o término do comando anterior.

- **Estrutura de decisão**

- O fluxo de instrução a ser seguido é escolhido em função do resultado da avaliação de uma ou mais condições.

- **Estrutura de repetição**

- Tem por objetivo repetir um trecho de programa um certo número de vezes. É também chamada de laço.

Questão 6 - Aula Exercícios 1

Dados o termo inicial e a razão de uma PA, bem como um número inteiro positivo n , calcule e escreva o valor do n -ésimo termo dessa PA

Solução

Algoritmo “ calculo n-esimo termo de uma PA”

inteiro a1, an, r, n

Inicio

escreva (“Entre com o primeiro termo, a razao e n”)

Leia(a1,r,n)

$an \leftarrow a1 + r * (n-1)$

escreva (“o n-esimo termo da PA e:”, An)

fim

fimalgoritmo

Questão

Dado um número inteiro com 3 dígitos decimais, isto é, um número inteiro positivo da forma xyz tal que x é um dígito de 1 a 9 e tanto y quanto z são dígitos de 0 a 9, obtenha e escreva a centena, dezena e unidade do número dado.

Por exemplo, se o número dado for 147, a solução do problema é 1,4 e 7. Observe que a entrada do problema consiste de um único valor, que é um número inteiro positivo representando uma centena, e não os três dígitos da centena

Solução

Algoritmo “ Obtem digitos de um numero””

inteiro numero, unidade, dezena,centena

Inicio

escreva (“Entre com um numero inteiro de 3 digitos”)

Leia(numero)

unidade <- numero % 10

numero <- numero / 10

dezena <- numero % 10

numero <- numero /10

centena <- numero % 10

escreva (“Centena:”, centena, “Dezena:”, dezena, “Unidade:”,
unidade)

fim

fimalgoritmo

Estruturas de decisão

- O fluxo de instrução a ser seguido é escolhido em função do resultado da avaliação de uma ou mais condições.
- Classificação quanto ao número de condições:
 - Uma condição (**decisão simples**): estrutura do **SE**.
 - Várias condições (**decisão múltipla**): estrutura do **ESCOLHA**.

Estrutura de decisão do tipo SE

- Sintaxe:

Simples	Composta
se (< condição >) então < comando 1 > fimse	se (< condição >) então < comando 1 > senão < comando 2 > fimse

Estrutura de decisão do tipo SE

- Semântica:
 - A condição que é uma expressão lógica é avaliada.
 - Se o resultado for verdadeiro então comando1 é executado e o fluxo do algoritmo prossegue com o primeiro comando após o **fimse**.
 - Se o resultado for falso, então comando2 é executado e, ao término do mesmo, o fluxo de execução prossegue com o primeiro comando após **fimse**.
 - Há casos em que senão comando2 é omitido.
 - Dessa forma, quando a condição é falsa, o fluxo de execução prossegue normalmente para a primeira instrução após o **fimse**, como se o comando **se** não existisse.
 - Os termos comando1 e comando2 podem conter um único comando, simples ou composto, ou mais de um.

Exemplo de uma estrutura de decisão simples

```
algoritmo "Verifica aprovação"  
  real n1, n2, média  
  início  
    escreva("Digite a primeira nota: ")  
    leia( n1 )  
    escreva("Digite a segunda nota: ")  
    leia( n2 )  
    media <- ( n1 + n2 ) / 2  
    se ( media >= 7.0 ) então  
      escreva ( "Aprovado(a)")  
    fimse  
  fim  
finalgoritmo
```

Exemplo de uma estrutura de decisão composta

algoritmo "Verifica aprovação ou reprovação"

real n1, n2, média

início

 escreva ("Digite a primeira nota: ")

 leia (n1)

 escreva ("Digite a segunda nota: ")

 leia (n2)

 media <- (n1 + n2) / 2

 se (media >= 7.0) então

 escreva ("Aprovado(a)")

 senão

 escreva ("Reprovado(a)")

 fimse

fim

finalgoritmo

Estrutura de decisão aninhada ou encaixada

- Usada quando uma alternativa envolver outras decisões.

Exemplo de uma estrutura de decisão aninhada ou encaixada

algoritmo "Verifica situação do(a) aluno(a)"

inteiro faltas, limfaltas

real n1, n2, n3, media

início

limfaltas <- 15 //para disciplinas c/ CH = 60h

// Entrada de dados

escreva ("Digite a primeira nota: ")

leia (n1)

escreva ("Digite a segunda nota: ")

leia (n2)

escreva ("Digite a terceira nota: ")

leia (n3)

escreva ("Digite a quantidade de
faltas: ")

leia (faltas)

media <- (n1 + n2 + n3) / 3

// Saída de dados

escreval (" _____ ")

escreval ("Resultado")

escreval (" _____ ")

escreval ("O limite de faltas permitido é de ",
limfaltas)

escreva ("A média do(a) aluno(a) é ", media)

se (media >= 7.0) ou (media >= 5 e n1>=3 e n2>=3
e n3>=3)

então

se (faltas <= limfaltas)

então

escreval (" - Aprovado(a)")

senão

escreval (" - Reprovado(a) por faltas")

fimse

senão

se (media < 3.0 ou faltas > limfaltas) então

escreval (" - Reprovado(a)")

senão

escreval (" - Prova substitutiva ")

fimse

fimse

fim

fimalgoritmo

Exercícios

- Faça um algoritmo que:
 1. Determine a mais alta de duas pessoas.
 2. Implemente uma calculadora com as operações de adição, subtração, multiplicação e divisão.

Exercícios

- Escreva um algoritmo que: leia as notas (de 0 a 10) de três provas de um aluno, calcule a média ponderada das três notas do aluno e escreva o *status dele como saída*. O *status do aluno* é “aprovado” se a média das notas é igual ou maior do que 7, “exame” se a média é igual ou maior do que 3, mas menor do que 7, e “reprovado” se a média é menor do que 3.

Exercícios

- Escreva um algoritmo que leia um número inteiro positivo com quatro dígitos e escreva “sim” se a soma dos algarismos da centena e milhar do número é par e “não” caso contrário.
- Escreva um algoritmo que receba 3 números inteiros e os apresente de forma ordenada crescente.