Algoritmo "Desenvolvimento\_06"

// Disciplina : [Linguagem e Lógica de Programação]

// Professor : Antonio Carlos Nicolodi

// Após uma prova de vestibular, uma escola quer identificar quantos dos seus

// alunos foram aprovados em cada turma de formandos. Ela quer saber qual

// aluno teve a maior nota e de qual turma ele é. Nessa escola, há

// quatro turmas (A, B, C e D) com vinte e cinco alunos cada.

// Elabore um algoritmo que solicite o nome e a nota do vestibular aos alunos. Depois, informe quantos deles foram aprovados, de que turma são, qual a maior nota em cada turma e qual aluno teve a maior nota de todas.

// Para que o aluno seja aprovado no vestibular, ele deverá obter nota maior

// ou igual a 7.

// ATENÇÃO: As notas por turma não podem se repetir.

Var

// Seção de Declarações das variáveis

i : inteiro

soma\_A : real

soma\_B : real

soma\_C : real

soma\_D : real

aprovados\_A : inteiro

aprovados\_B : inteiro

aprovados\_C : inteiro

aprovados\_D : inteiro

media\_A : real

media\_B : real

media\_C : real

media\_D : real

nomes\_turmaA : vetor [1..25] de caracter

notas\_turmaA : vetor [1..25] de real

nomes\_turmaB : vetor [1..25] de caracter

notas\_turmaB : vetor [1..25] de real

nomes\_turmaC : vetor [1..25] de caracter

notas\_turmaC : vetor [1..25] de real

Inicio

// Seção de Comandos, procedimento, funções, operadores, etc...

aprovados\_A <- 0

aprovados\_B <- 0

aprovados\_C <- 0

aprovados\_D <- 0

media\_A <- 0.0

media\_B <- 0.0

media\_C <- 0.0

media\_D <- 0.0

escreval("Digite os Nomes e a Notas da Turma A ")

para i de 1 ate 25 faca

escreval("Digite o Nome: ")

leia(nomes\_turmaA[i])

escreval("Digite a Nota: ")

leia(notas\_turmaA[i])

se (notas\_turmaA[i] >= 7.0)entao

aprovados\_A <- aprovados\_A +1

fimse

soma\_A <- soma\_A + notas\_turmaA[i]

fimpara

media\_A <- soma\_A / 25

escreval("Turma A Aprovados: ",aprovados\_A)

escreval("Turma A Media: ",media\_A)

escreval("----------------------------------------")

escreval("Digite os Nomes e a Notas da Turma B ")

para i de 1 ate 25 faca

escreval("Digite o Nome: ")

leia(nomes\_turmaB[i])

escreval("Digite a Nota: ")

leia(notas\_turmaB[i])

se (notas\_turmaB[i] >= 7.0) entao

aprovados\_B <- aprovados\_B +1

fimse

soma\_B <- soma\_B + notas\_turmaB[i]

fimpara

media\_B <- soma\_B / 25

escreval("Turma B Aprovados: ",aprovados\_B)

escreval("Turma B Media: ",media\_B)

escreval("----------------------------------------")

escreval("Digite os Nomes e a Notas da Turma C ")

para i de 1 ate 25 faca

escreval("Digite o Nome: ")

leia(nomes\_turmaC[i])

escreval("Digite a Nota: ")

leia(notas\_turmaC[i])

se (notas\_turmaC[i] >= 7.0) entao

aprovados\_C <- aprovados\_C +1

fimse

soma\_C <- soma\_C + notas\_turmaC[i]

fimpara

media\_C <- soma\_C / 25

escreval("Turma C Aprovados: ",aprovados\_C)

escreval("Turma C Media: ",media\_C)

escreval("----------------------------------------")

escreval("Digite os Nomes e a Notas da Turma D ")

para i de 1 ate 25 faca

escreval("Digite o Nome: ")

leia(nomes\_turmaD[i])

escreval("Digite a Nota: ")

leia(notas\_turmaD[i])

se (notas\_turmaD[i] >= 7.0) entao

aprovados\_D <- aprovados\_D +1

fimse

soma\_D <- soma\_D + notas\_turmaD[i]

fimpara

media\_D <- soma\_D / 25

escreval("Turma D Aprovados: ",aprovados\_D)

escreval("Turma D Media: ",media\_D)

Fimalgoritmo