

Bacharelado em Sistemas de Informação
SIF001 – Algoritmos. Aula 10 - 12/05/2020
Professores Antonello e Antero

Exercícios

- 1) Além de contar seus carros de luxo, um conhecido professor de matemática tem como hobby acompanhar corridas de lesmas.

A corrida de lesmas é um esporte que cresceu muito nos últimos anos, fazendo com que várias pessoas dediquem suas vidas tentando capturar lesmas velozes, e treiná-las para faturar milhões em corridas pelo mundo. Porém a tarefa de capturar lesmas velozes não é uma tarefa muito fácil, pois praticamente todas as lesmas são muito lentas. Cada lesma é classificada em um nível dependendo de sua velocidade:

- a) Nível 1: Se a velocidade é menor que 10 cm/h .
- b) Nível 2: Se a velocidade é maior ou igual a 10 cm/h e menor que 20 cm/h .
- c) Nível 3: Se a velocidade é maior ou igual a 20 cm/h .

Sua tarefa é desenvolver um programa, que use sub-rotina, para ajudar o digníssimo professor a identificar qual nível de velocidade da lesma mais veloz de um grupo de lesmas.

Entrada: A entrada ocorre no programa principal e consiste de 5 número inteiros que representam a velocidade de cada uma das lesmas do grupo.

Saída: A saída de dados deve ser exibida no programa principal e deve ser composta por três números: A velocidade da lesma vencedora; o nível da lesma vencedora; velocidade média das 5 lesmas do grupo.

```
funcao CorridaLesma(A,B,C,D,F: inteiro; Var velMax: inteiro; Var velMed:
real): inteiro
var
    Aux: inteiro
    Nivel: inteiro

inicio
    velMax <- A
    Se (velMax < B) entao
        velMax <- B
    FimSe
    Se (velMax < C) entao
        velMax <- C
    FimSe
    Se (velMax < D) entao
        velMax <- D
    FimSe
    Se (velMax < F) entao
        velMax <- F
    FimSe

    VelMed <- (A + B + C + D + F)/5

    Se (velMax < 10) entao
```

```

        Nivel <- 1
    Senao
        Se (velMax > 10) e (velMax < 20) entao
            Nivel <- 2
        Senao
            Nivel <- 3
        FimSe
    Fimse

    retorne Nivel
fimfuncao

```

algoritmo "Lesmas"

```

var
    L1, L2, L3, L4, L5, velVenc, nivVenc, velMedia: inteiro

inicio
    Leia (L1, L2, L3, L4, L5)
    nivVenc <- CorridaLesma(L1,L2,L3,L4,L5,VelVenc,VelMedia)

    Escreval ("Velocidade da lesma vencedora:", velVenc)
    Escreval ("Nível da lesma vencedora:", nivVenc)
    Escreval ("Velocidade média:", velMedia)

finalgoritmo

```

2) Fonte: baseado em URI Online Judge | 1214

No início da graduação sabe-se que 90% dos calouros tem sempre a expectativa de irem bem no curso e alguns deles querem estar acima da média da turma. Baseado em uma série de notas de calouros do ano passado, o professor de matemática pede para você fazer um programa que ajude a entender se a expectativa dos alunos condiz com a realidade. Use função ou procedimento para verificar a porcentagem de alunos acima da média.

Entrada

A primeira linha da entrada de dados corresponde ao número (N) de turmas a serem analisadas.

Outras N linhas representam N turmas de 5 alunos. Assim, cada linha contém as notas médias de 5 alunos.

Saída

A saída de dados deve ocorrer no programa principal e para cada turma analisada imprima uma linha dando o percentual de estudantes que estão igual ou acima da média da turma, com o valor arredondado e com 1 casa decimal.

```
funcao CalcMedia (n1,n2,n3,n4,n5: real): real
```

```
var
```

```
    media, porcentagem: real
```

```
    cont: inteiro
```

```
inicio
```

```
    media <- (n1+n2+n3+n4+n5)/5
```

```
    cont<-0
```

```
    se (n1 > media) entao
```

```
        cont <- cont +1
```

```
    fimse
```

```
    se (n2 > media) entao
```

```
        cont <- cont +1
```

```
    fimse
```

```
    se (n3 > media) entao
```

```
        cont <- cont +1
```

```
    fimse
```

```
    se (n4 > media) entao
```

```
        cont <- cont +1
```

```
    fimse
```

```
    se (n5 > media) entao
```

```
        cont <- cont +1
```

```
    fimse
```

```
    porcentagem <- cont / 5 * 100
```

```
    retorne porcentagem
```

```
fimfuncao
```

```
algoritmo "Exerc02"
```

```
var
```

```
    nota1, nota2, nota3, nota4, nota5, p: real
```

```
    N, i: inteiro
```

```
inicio
```

```
    escreva ("Quantidade de turmas:")
```

```
    leia(N)
```

```
    Para i de 1 Ate N Passo 1 Faca
```

```
        escreva("Nota do primeiro aluno:")
```

```
        leia(nota1)
```

```
        escreva("Nota do segundo aluno:")
```

```
        leia(nota2)
```

```
        escreva("Nota do terceiro aluno:")
```

```
        leia(nota3)
```

```
        escreva("Nota do quarto aluno:")
```

```
        leia(nota4)
```

```
        escreva("Nota do quinto aluno:")
```

```
        leia(nota5)
```

```
    p <- CalcMedia (nota1, nota2, nota3, nota4, nota5)
```

```
    Escreval("Porcentagem de alunos acima da média da turma:",  
p:1:1,"%")
```

```
    FimPara
```

```
fimalgoritmo
```

- 3) Alterar o código desenvolvido anteriormente, para use de sub-rotina.

Fonte: URI Online Judge | 1015

<https://www.urionlinejudge.com.br/judge/pt/problems/view/1015>

Leia os quatro valores correspondentes aos eixos x e y de dois pontos quaisquer no plano, p1(x1,y1) e p2(x2,y2) e calcule a distância entre eles, mostrando 4 casas decimais após a vírgula, segundo a fórmula:

$$\text{Distancia} = \sqrt{(x2 - x1)^2 + (y2 - y1)^2}$$

Entrada

O arquivo de entrada contém duas linhas de dados. A primeira linha contém dois valores de ponto flutuante: x1 y1 e a segunda linha contém dois valores de ponto flutuante x2 y2.

Saída

Calcule e imprima o valor da distância segundo a fórmula fornecida, com 4 casas após o ponto decimal.

Exemplo de Entrada	Exemplo de Saída
1.0 7.0 5.0 9.0	4.4721

```
Funcao DistanciaPontos(A, B, C, D: real): real
```

```
Var
```

```
    distancia: real
```

```
Inicio
```

```
    distancia <- ((C - A)^2) + ((D - B)^2)
```

```
    distancia <- Raizq(distancia)
```

```
    retorne distancia
```

```
FimFuncao
```

```
Algoritmo "EX03"
```

```
Var
```

```
    p1x, p1y, p2x, p2y, dist: real
```

```
Inicio
```

```
    Escreval("Coordenadas X e Y do ponto 1:")
```

```
    Leia(p1x, p1y)
```

```
    Escreval("Coordenadas X e Y do ponto 2:")
```

```
    Leia(p2x, p2y)
```

```
    dist <- DistanciaPontos(p1x, p1y, p2x, p2y)
```

```
    Escreval("Distância calculada: ", dist:6:4)
```

```
FimAlgoritmo
```