

Algoritmos e Programação I

Problemas e Soluções

Prof. Fernando Maia da Mota
mota.fernandomaia@gmail.com
CPCX/UFMS
© Fernando Maia da Mota

Problemas e Soluções

❖ Nesta aula, veremos alguns problemas e suas respectivas soluções.

Problemas e Soluções

- ❖ Em uma turma de algoritmos formada por 4 alunos, o professor deseja saber se o número de aprovações na turma é par ou ímpar. Sabendo-se que a média de aprovação da disciplina é 6 construa um algoritmo que leia as notas finais destes alunos e verifique se o número de aprovações da turma é par ou ímpar.

INPUT	OUTPUT
2, 3, 1, 6	QTD ímpar de alunos aprovados
1, 7, 7, 2	QTD par de alunos aprovados
2, 7, 6, 8	QTD ímpar de alunos aprovados

Problemas e Soluções

```
// algoritmo que verifica se número de aprovados é par ou impar
algoritmo "verifica se # aprovados é par ou impar"
    // declaração de variáveis
    var
        a, b, c, d : inteiro
    inicio

        // lê as notas finais
        leia a, b, c, d

        se (a >= 6 XOR b >= 6 XOR c >= 6 XOR d >= 6)
            então
                escreva "QTD impar de alunos aprovados"
            senão
                escreva "QTD par de alunos aprovados"
    fimalgoritmo
```

Problemas e Soluções

- ❖ Uma pessoa aplicou seu capital a juros e deseja saber, trimestralmente, a posição de seu investimento inicial c . Chamando de i a taxa de juros do trimestre, escrever uma tabela que forneça, para cada trimestre, o rendimento auferido e o saldo acumulado durante o período de x anos, supondo que nenhuma retirada tenha sido feita.

Problemas e Soluções

INPUT	OUTPUT
Capital 100, Juros 3.7, Anos 1	1 : 3.70
	1 : 103.70
	2 : 3.83
	2 : 107.53
	3 : 3.97
	3 : 111.50
	4 : 4.12
	4 : 115.62

Problemas e Soluções

// algoritmo para calcular rendimento e montante de aplicação trimestral

algoritmo "calcula rendimento"

// declaração de variáveis

var

x, n, j : inteiro

c, i, r : real

inicio

// lê o capital inicial, a taxa de juros e números de anos

leia c, i, x

// calcula o número de trimestres em x anos

n ← x * 4

// calcula e exhibe rendimento e montante

para j de 1 até n faça

r ← (i * c) / 100

c ← c + r

escreva "rendimento do trimestre ", j, ": ", r

escreva "montante do trimestre ", j, ": ", c

fimpara

fimalgoritmo

Problemas e Soluções

- ❖ Em um frigorífico existem 90 bois. Cada boi traz em seu pescoço um cartão contendo seu número de identificação e seu peso. Faça um algoritmo que encontre e escreva o número e o peso do boi mais gordo e do boi mais magro.

INPUT	OUTPUT
1 , 260.5	O número do boi mais gordo é: 1
2 , 262	O peso do boi mais gordo é: 262
3 , 257.9	O número do boi mais magro é: 3
4 , 259.7	O peso do boi mais magro é: 257.9

Problemas e Soluções

```
// algoritmo para encontrar o número e o peso do boi mais gordo e
// do boi mais magro de um conjunto de 90 bois
algoritmo "encontra número e peso"
```

```
// declaração de variáveis
```

```
var
```

```
    num, boigordo, boimagro : inteiro
    peso, maiorpeso, menorpeso : real
```

```
inicio
```

```
    // lê os dados do primeiro boi
    escreva "entre com o número de identificação do primeiro boi: "
    leia num
    escreva "entre com o peso do primeiro boi: "
    leia peso
```

```
    // inicializa as variáveis que conterão o resultado
    boigordo ← num
    boimagro ← num
    maiorpeso ← peso
    menorpeso ← peso
```

Problemas e Soluções

// compara os pesos para encontrar o boi mais gordo e o boi mais magro
para j de 2 até 90 faça

escreva "entre com o número do próximo boi: "

leia num

escreva "entre com o peso do próximo boi: "

leia peso

// compara o peso lido com o maior peso até então

se peso > maiorpeso então

maiorpeso ← peso

boigordo ← num

senão

se peso < menorpeso então

menorpeso ← peso

boimagro ← num

fimse

fimse

fimpara

// escreve o número e o peso do boi mais gordo e do boi mais magro

escreva "o número do boi mais gordo é: ", boigordo

escreva "o peso do boi mais gordo é: ", maiorpeso

escreva "o número do boi mais magro é: ", boimagro

escreva "o peso do boi mais magro é: ", menorpeso

fimalgoritmo

Problemas e Soluções

- ❖ Uma pesquisa sobre algumas características físicas da população de uma determinada região coletou os seguintes dados, referentes a cada habitante, para serem analisados:
 - idade em anos
 - sexo (masculino, feminino)
 - cor dos olhos (azuis, verdes, castanhos)
 - cor dos cabelos (louros, castanhos, pretos)
- ❖ Para cada habitante são informados os quatro dados acima. A fim de indicar o final da entrada, após a sequência de dados dos habitantes, o usuário entrará com o valor -1 para a idade, o que deve ser interpretado pelo algoritmo como fim de entrada.

Problemas e Soluções

- ❖ Seu algoritmo deve encontrar a maior idade do conjunto de indivíduos e o percentual de indivíduos do sexo feminino com idade entre 18 e 35 anos, inclusive, e olhos verdes e cabelos louros.
- ❖ Considere o uso de cadeias na criação deste algoritmo.
- ❖ Observe que este algoritmo possui um **critério** de parada.
 - ❖ “[...] o usuário entrará com o valor -1 para a idade, o que deve ser interpretado pelo algoritmo como fim de entrada.”

Problemas e Soluções

```
// algoritmo para encontrar a maior idade de um conjunto de indivíduos
// e o percentual de indivíduos do sexo feminino com idade entre 18 e
// 35 anos, inclusive, e olhos verdes e cabelos louros
algoritmo "encontra maior idade e percentual"
```

```
// declaração de variáveis
```

```
var
```

```
idade, maioridade, habitantes, totalhabitantes : inteiro
percentagem : real
sexo, olhos, cabelos : cadeia
```

```
inicio
```

```
// inicializa algumas variáveis
```

```
maioridade ← -1
```

```
habitantes ← 0
```

```
totalhabitantes ← 0
```

```
// lê a idade do primeiro habitante
```

```
escreva "entre com a idade do habitante ou -1 para encerrar: "
```

```
leia idade
```

Problemas e Soluções

```
// calcula os resultados
enquanto idade ≠ -1 faça
    escreva "entre com o sexo do habitante: "
    leia sexo

    escreva "entre com a cor dos olhos do habitante: "
    leia olhos

    escreva "entre com a cor dos cabelos do habitante: "
    leia cabelos

// compara a idade lida com a maior idade até então
se (idade > maioridade) então
    maioridade ← idade

fimse

// conta o número total de habitantes
totalhabitantes ← totalhabitantes + 1
```

Problemas e Soluções

restrições

cabelos louros)

se (idade \geq 18 E idade \leq 35) E (sexo ="feminino") E (olhos ="verdes") E (cabelos ="louros") então

habitantes \leftarrow habitantes + 1

fimse

// lê a idade do próximo habitante

escreva "entre com a idade do habitante ou -1 para encerrar: "

leia idade

fimenquanto

// escreve a maior idade e a porcentagem pedida

se totalhabitantes $>$ 0 então

porcentagem \leftarrow (habitantes/totalhabitantes) * 100

escreva "a maior idade é: ", maioridade

escreva "a porcentagem é: ", porcentagem

fimse

fimalgoritmo

Referências

- ❖ **SIQUEIRA, Marcelo F. Algoritmos e Estrutura de Dados. Mato Grosso do Sul: CCET/CPCX - UFMS, 2007.**