

Bacharelado em Sistemas de Informação  
SIF001 – Algoritmos. Aula 07 - 14/04/2020  
Professores Antonello e Antero

**Bibliografia básica desta aula**

- ✓ Medina & Fertig (2006). Capítulo 05.
- ✓ Forbellone & Eberspacher (2000). Capítulo 06.
- ✓ Ascencio & Campos (2002). Capítulo 10.

**Exercícios**

1) Ovos de páscoa (entregar)

Uma entidade assistencial recebeu R\$1.500,00 em doações para compra de ovos de Páscoa. Um funcionário dessa entidade tem no carrinho de compras uma quantidade de ovos, de vários tipos e tamanhos, que se somados os valores provavelmente passe do montante doado. Então, alguns desses ovos vão ficar no supermercado.

Acontece que o caixa do supermercado também que participar desta empreitada e disse que completará o valor necessário de apenas um item, logo que o valor da compra passar do montante que a entidade tem para gastar.

Faça um algoritmo que possibilite ir entrando, um a um, com vários itens comprados, de tal forma que se interrompa conforme as condições estabelecidas (cada item tem a quantidade e preço individual).

Informar quantos ovos serão comprados e quanto de dinheiro o caixa do supermercado vai completar para o pagamento.

```
algoritmo "A07Ex01"
```

```
var
```

```
    preco, total, restante: real  
    quantidade, qntdTotal: inteiro
```

```
inicio
```

```
total <- 0
```

```
qntdTotal <- 0
```

```
repita
```

```
    escreva("Digite o valor do produto: ")
```

```
    leia(preco)
```

```
    escreva("Digite a quantidade do produto: ")
```

```
    leia(quantidade)
```

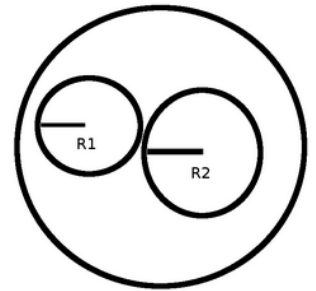
```
    total <- total + (preco*quantidade)
    qntdTotal <- qntdTotal + quantidade
ate(total >= 300)

restante <- total - 300
escrevaL("Quantidade de produtos comprados: ", qntdTotal)
escrevaL("O Prof. Camilo vai pagar: R$", restante)
fimalgoritmo
```

## 2) Conduites (entregar)

Você está prestando serviço para um escritório de engenharia civil e precisa definir os diâmetros de vários conduítes a serem instalados em uma determinada obra.

- Em cada conduíte passará sempre dois cabos de energia, conforme a figura ao lado.
- Determine e exiba o diâmetro mínimo de um conduíte a partir do raio dos dois cabos estabelecidos na entrada de dados.
- O processamento se encerra quando pelo menos um raio entrado for igual a zero.



```
algoritmo "A07Ex02"
```

```
// Solução proposta por integrante da turma de alunos
```

```
var
```

```
    raio1,raio2,cond,ctg:real
```

```
inicio
```

```
    escreva("Raio do primeiro fio:")
```

```
    leia(raio1)
```

```
    escreva("Raio do segundo fio:")
```

```
    leia(raio2)
```

```
    enquanto (raio1<>0) E (raio2<>0) faca
```

```
        cond <- 2*(raio1+raio2)
```

```
        Escreval("Diâmetro mínimo do conduíte", cond)
```

```
        escreva("Raio do primeiro fio:")
```

```
        leia(raio1)
```

```
        escreva("Raio do segundo fio:")
```

```
        leia(raio2)
```

```
    fimenquanto
```

```
fimalgoritmo
```

```
algoritmo "A07Ex02"
```

```
// Solução proposta por integrante da turma de alunos
```

```
var
```

```
    diametroConduite, raio1, raio2 :real
```

```
inicio
```

```
    Repita
```

```
        Escreval("Informe o raio do primeiro cabo")
```

```
        Leia(raio1)
```

```
        Escreval("Informe o raio do segundo raio cabo")
```

```
        Leia(raio2)
```

```
        diametroConduite<-raio1*2+raio2*2
```

```
        Escreval("Diâmetro mínimo do conduíte", diametroConduite)
```

```
    Ate (raio1<=0) ou (raio2<=0)
```

```
fimalgoritmo
```

3) URI 1117 - Validação de Nota (entregar)

<https://www.urionlinejudge.com.br/judge/pt/problems/view/1117>

Faça um programa que leia as notas referentes às duas avaliações de um aluno. Calcule e imprima a média semestral. Faça com que o algoritmo só aceite notas válidas (uma nota válida deve pertencer ao intervalo  $[0,10]$ ). Cada nota deve ser validada separadamente.

Entrada

A entrada contém vários valores reais, positivos ou negativos. O programa deve ser encerrado quando forem lidas duas notas válidas.

Saída

Se uma nota inválida for lida, deve ser impressa a mensagem "nota invalida". Quando duas notas válidas forem lidas, deve ser impressa a mensagem "media = " seguido do valor do cálculo. O valor deve ser apresentado com duas casas após o ponto decimal.

```
algoritmo "URI1117"
var
    N1, N2, media: real
inicio
    escreval("Digite a primeira nota")
    Leia(N1)
    Enquanto (N1<0) OU (N1>10) Faca
        Escreval("Primeira nota inválida, digite novamente")
        Leia(N1)
    FimEnquanto

    escreval("Digite a segunda nota")
    Leia(N2)
    Enquanto (N2<0) OU (N2>10) Faca
        Escreval("Segunda nota inválida, digite novamente")
        Leia(N2)
    FimEnquanto

    media <- (N1+N2)/2;
    Escreval("media = ", media:1:2)
FimAlgoritmo
```

4) URI 1134 – Tipo de combustível (entregar)

<https://www.urionlinejudge.com.br/judge/pt/problems/view/1134>

Um Posto de combustíveis deseja determinar qual de seus produtos tem a preferência de seus clientes. Escreva um algoritmo para ler o tipo de combustível abastecido (codificado da seguinte forma: 1.Álcool 2.Gasolina 3.Diesel 4.Fim). Caso o usuário informe um código inválido (fora da faixa de 1 a 4) deve ser solicitado um novo código (até que seja válido). O programa será encerrado quando o código informado for o número 4.

Entrada

A entrada contém apenas valores inteiros e positivos.

Saída

Deve ser escrito a mensagem: "MUITO OBRIGADO" e a quantidade de clientes que abasteceram cada tipo de combustível, conforme exemplo.

Exemplo de Entrada	Exemplo de Saída
8 1 7 2 2 4	MUITO OBRIGADO Alcool: 1 Gasolina: 2 Diesel: 0

```
algoritmo "URI1134"
```

```
// Solução proposta por integrante da turma de alunos
```

```
var
```

```
    qntdAlcool, qntdGasolina, qntdDiesel, tipoComb: inteiro
```

```
inicio
```

```
repita
```

```
    leia(tipoComb)
```

```
    escolha(tipoComb)
```

```
        caso 1
```

```
            qntdAlcool <- qntdAlcool + 1
```

```
        caso 2
```

```
            qntdGasolina <- qntdGasolina + 1
```

```
        caso 3
```

```
            qntdDiesel <- qntdDiesel + 1
```

```
        caso 4
```

```
            escrevaL("MUITO OBRIGADO")
```

```
            escrevaL("Alcool:", qntdAlcool)
```

```
            escrevaL("Gasolina:", qntdGasolina)
```

```
            escrevaL("Diesel:", qntdDiesel)
```

```
        outrocaso
            escrevaL("Tipo de combustível não disponível, digite
                    novamente")
    fimescolha

ate (tipoComb = 4)

finalgoritmo
```