

ALGORÍTMOS DE LÓGICA DE PROGRAMAÇÃO 1

52/2021

MAPA

Desta forma, em seu PSEUDOCÓDIGO, você deverá utilizar a estrutura de dados heterogêneas para armazenar em memória as especificações de um celular. De acordo com Leal e Oliveira (2020, p. 157) “As estruturas de dados heterogêneas permitem o agrupamento de informações de diferentes tipos de dados, sendo denominadas registros”.

Diante do exposto, você precisa desenvolver seu **algoritmo em pseudocódigo** e atender aos seguintes critérios:

- 1 – Estrutura de Registro para armazenar 100 aparelhos celulares
- 2 – Um menu que apresente os seguintes módulos (procedimentos / funções)
 - 2.1 – Cadastrar Produtos
 - 2.2 – Listar todos (Aqui liste 3 características armazenadas)
 - 2.3 - Listar 1 produto específico (Aqui liste 3 características armazenadas)

Considere as seguintes regras:

- 1 - Controlar o cadastro para que não haja códigos idênticos.
- 2 - Informar ao usuário se produto foi cadastro ou não.
- 3 - Informar ao usuário que o produto não foi encontrado caso ocorra.

Algoritmo cadastro_de_produtos

```
Tipo produto = registro
    codigo: inteiro
    nome: vetor[1..50 ] de caractere
    quantidade: inteiro
    valor: real
Fim_registro

Função selecionaMenu():inteiro
    Var
        menu: inteiro
Início
    Escreval("")
    Escreval("SELECIONE O MENU DESEJADO")
    Escreval("")
    Escreval(" 1 – CADASTRAR PRODUTOS")
    Escreval(" 2 – LISTAR TODOS")
    Escreval(" 3 – LISTAR 1 PRODUTO ESPECÍFICO ")
    Escreval(" 4 – SAIR")
    Escreval("")
    Escreva("  DIGITE UMA OPÇÃO: ")
    Leia(menu)
    Escreval("")
    Retorne menu
    Limpa_tela
Fim_função
```

//Regra 1 - Controlar o cadastro para que não haja códigos idênticos.

```
Função existeCodigo(cod: inteiro): lógico
  Var
    aux: inteiro
  Início
    Retorne falso
    Para aux de 1 até índice passo 1 faça
      Se (listaProdutos[aux].codigo = cod) então
        Retorne verdadeiro
      Fim_se
    Fim_para
  Fim_função
```

// 2.1 – Cadastrar Produtos

```
Procedimento cadastrarProduto()
  Var
    prod: produto
    aux: inteiro
    achou: lógico
  Início
    Escreval("VOCÊ ESCOLHEU: 1 – CADASTRAR PRODUTOS ")
    Escreval("")
    Se (índice < 100) então
      índice <- índice + 1
      Repita
        Escreva("  Digite o código: ")
        Leia(prod.codigo)
        achou <- existeCodigo(prod.codigo)
        Se(achou = verdadeiro) então
          Limpa_tela
          Escreval("CÓDIGO EXISTENTE, TENDE OUTRO!")
          Escreval("")
        Fim_se
      Até_que(achou = falso)

      Escreva("  Digite o nome: ")
      Leia(prod.nome)
      Escreva("  Digite o valor: ")
      Leia(prod.valor)
      Escreva("  Digite a Quantidade: ")
      Leia(prod.quantidade)
      listaProdutos[índice] <- prod
```

//Regra 2 - Informar ao usuário se produto foi cadastro ou não.

```
  Limpa_tela
  Escreval("PRODUTO CADASTRADO COM SUCESSO !")
```

```

Senão
    Escreval("FALHA AO CADASTRAR ! LIMITE ATINGIDO")
Fim_se
Fim_procedimento

// 2.2 – Listar todos (Aqui liste 3 características armazenadas)
Procedimento listarTodos()
    Var
        aux: inteiro
    Início
        Escreval("VOCÊ ESCOLHEU: 2 – LISTAR TODOS ")
        Escreval("")
        Se (indice > 0) então
            Para aux de 1 ate indice passo 1 faça
                Escreval("  Código ",listaProdutos[aux].codigo)
                Escreval("  Nome ",listaProdutos[aux].nome)
                Escreval("  Valor R$ ",listaProdutos[aux].valor)
                Escreval("  Quantidade ",listaProdutos[aux].quantidade)
                Escreval("")
            Fim_para
        Senão
            Escreval("NENHUM PRODUTO FOI CADASTRADO AINDA !")
        Fim_se

    Fim_procedimento

```

```

//2.3 - Listar 1 produto específico (Aqui liste 3 características armazenadas)
Procedimento listarProdutoEspecifico()
    Var
        aux: inteiro
        cod: inteiro
        achou: logico
    Início
        Escreval("VOCÊ ESCOLHEU: 3 – LISTAR 1 PRODUTO ESPECÍFICO ")
        Escreval("")

        Se (indice > 0) então
            Escreva("  Digite o código do produto: ")
            Leia(cod)

            achou <- falso
            Para aux de 1 ate indice passo 1 faça
                Se (listaProdutos[aux].codigo = cod) então
                    Escreval("")
                    Escreval("PRODUTO ENCONTRADO")
                    Escreval("")
                    Escreval("  Código ",listaProdutos[aux].codigo)
                    Escreval("  Nome ",listaProdutos[aux].nome)
                    Escreval("  Valor R$ ",listaProdutos[aux].valor)

```

```

        Escreval("  Quantidade ",listaProdutos[aux].quantidade)
        Escreval("")
        achou <- verdadeiro
        Fim_se
    Fim_para

//Regra 3 - Informar ao usuário que o produto não foi encontrado caso ocorra.
    Se (achou = falso) então
        Escreval("")
        Escreval("PRODUTO NÃO ENCONTRADO !")
        Fim_se

    Senão
        Escreval("NENHUM PRODUTO FOI CADASTRADO AINDA")
        Fim_se

    Fim_procedimento

Var
// 1 – Estrutura de Registro para armazenar 100 aparelhos celulares
listaProdutos: vetor[1..100] de produto
indice: inteiro
opcao: inteiro

Início
// 2 – Um menu que apresente os seguintes módulos (procedimentos / funções)
indice <- 0
Repita
    opcao <- selecionaMenu()

    Caso(opcao)
        Seja 1 faça
            cadastrarProduto()
        Seja 2 faça
            listarTodos()
        Seja 3 faça
            listarProdutoEspecifico()
        Seja 4 faça
            Escreval("FIM DO PROGRAMA")
        Senão
            Escreval("OPÇÃO INVÁLIDA")
        Fim_caso
    Até_que(opcao = 4)

Fim_algoritmo

```