

EXERCÍCIOS ALGORITMOS – VISUALG (AULA 6)

Algoritmo "De_100_ate_110"

// Descrição: Imprime os 10 primeiros números inteiros depois de 100

// Autor: Gabriel Henrique

// Data atual: 15/03/2024

Var

cont: inteiro

Inicio

escreval(" -| DE 100 ATÉ 110 TEMOS |-")

escreval

para cont de 101 ate 110 faca

se (cont > 101) e (cont <= 110) entao

escreva(" -")

fimse

escreva(cont)

fimpara

Fimalgoritmo

Algoritmo "Pares_1_ate_100"

// Descrição: Imprime os números pares de 1 até 100

// Autor: Gabriel Henrique

// Data atual: 15/03/2024

Var

cont: inteiro

Inicio

escreval(" «-- NÚMEROS PARES DE 1 A 100 --»")

escreval

para cont de 1 ate 100 faca

se (cont % 2 = 0) entao

escreval(" |",cont, " |")

fimse

fimpara

Fimalgoritmo

Algoritmo "Numero_Maximo"

// Descrição: Imprime determinada quantidade de números até exibir uma mensagem

// Autor: Gabriel Henrique

// Data atual: 15/03/2024

Var

cont, n: inteiro

Inicio

escreva(" - Digite quantos números quer imprimir -> ")

leia(n)

escreval

para cont de 1 ate n faca

se (cont > 30) entao

escreval

escreval

escreval(" * NÚMERO MÁXIMO ATINGIDO *")

interrompa

fimse

se (cont > 1) e (cont <= 30) entao

escreva (" |")

fimse

escreva(cont)

fimpara

Fimalgoritmo

Algoritmo "Multiplos_cinco"

// Descrição: Imprime os próximos múltiplos de 5 de um número informado

// Autor: Gabriel Henrique

// Data atual: 13/03/2024

Var

cont, n: inteiro

Inicio

escreval(" «--| MÚLTIPLOS DE CINCO |--»")

escreval

escreva(" * Informe um número -> ")

leia(n)

escreval

cont <- n + 1

enquanto (cont > n) faca

se (cont % 5 = 0) entao

escreval (" |->",cont, " «-|")

fimse

cont <- cont + 1

se (cont > (n + 50)) entao

interrompa

fimse

fimenquanto

Fimalgoritmo

Algoritmo "Contador_crescente"

// Descrição: Conta números pares e ímpares de forma crescente

// Autor: Gabriel Henrique

// Data atual: 13/03/2024

Var

cont, n: inteiro

opc: caractere

Inicio

escreval(" ^ CONTADOR CRESCENTE ^")

escreval

escreva(" * Digite um número -> ")

leia(n)

escreva(" * Par ou Ímpar (P/I)? -> ")

leia(opc)

opc <- maiusc(opc)

escreval

se (opc <> "P") e (opc <> "I") e (opc <> "PAR") e (opc <> "IMPAR") entao

 escreval(" DIGITE PAR OU IMPAR!")

 finalgoritmo

fimse

para cont de 0 ate n faca

 se (opc = "P") ou (opc = "PAR") entao

 se (cont % 2 = 0) entao

se (cont > 0) e (cont <= n) entao

 escreva("-|-")

fimse

escreva(cont)

se (cont > 1000) entao

 escreval

 escreval

 escreval(" VALOR MÁXIMO ATINGIDO!")

 interrompa

fimse

fimse

senao se (opc = "I") ou (opc = "IMPAR") entao

se (cont % 2 <> 0) entao

se (cont > 1) e (cont <= n) entao

 escreva("-|-")

fimse

escreva(cont)

se (cont > 1000) entao

 escreval

 escreval

 escreval(" VALOR MÁXIMO ATINGIDO!")

 interrompa

fimse

fimse

fimse

fimpara

Fimalgoritmo

Algoritmo "Contador_decrescente"

// Descrição: Conta números pares e ímpares de forma decrescente

// Autor: Gabriel Henrique

// Data atual: 13/03/2024

Var

cont, n: inteiro

opc: caractere

Inicio

escreval(" ^\ CONTADOR DECRESCENTE ^")

escreval

escreva(" * Digite um número -> ")

leia(n)

escreva(" * Par ou Ímpar (P/I)? -> ")

leia(opc)

opc <- maiusc(opc)

escreval

se (opc <> "P") e (opc <> "I") e (opc <> "PAR") e (opc <> "IMPAR") entao

 escreval(" DIGITE PAR OU IMPAR!")

 finalgoritmo

fimse

para cont de n ate 0 passo -1 faca

 se (opc = "P") ou (opc = "PAR") entao

 se (cont % 2 = 0) entao

se (cont >= 0) e (cont < (n - 1)) entao

 escreva("-|-")

fimse

 escreva(cont)

fimse

senao se (opc = "I") ou (opc = "IMPAR") entao

 se (cont % 2 <> 0) entao

 se (cont >= 1) e (cont < (n - 1)) entao

 escreva("-|-")

 fimse

 escreva(cont)

 fimse

fimse

fimpara

Fimalgoritmo

Algoritmo "Loja_calcados"

// Descrição: Usuário faz pedidos e é imprimido os produtos escolhidos e o valor total

// Autor: Gabriel Henrique

// Data atual: 14/03/2024

Var

cod: inteiro

opc, produtos: caractere

interacao: logico

valor: real

Inicio

interacao <- verdadeiro

opc <- "S"

repita

se (interacao = verdadeiro) entao

interacao <- falso

senao

limpatela()

fimse

escreval(" «---- SK STORE ----»")

escreval(" =====")

escreval(" | CÓDIGO | PRODUTOS | PREÇO |")

escreval(" =====")

escreval(" | 1 | Tênis | R\$ 249,99 |")

escreval(" | 2 | Sandália | R\$ 95,20 |")

escreval(" | 3 | Chinelo | R\$ 22,90 |")

```
escreval("      | 4 | Sapatilha | R$ 87,10 |")
escreval("      | 5 | Bota      | R$ 173,90 |")
escreval("=====")
escreval
```

```
escreva("  * Olá! Informe o código do produto que deseja -> ")
leia(cod)
escreval
```

```
se (cod < 1) ou (cod > 5) entao
```

```
    enquanto (cod < 1) ou (cod > 5) faca
```

```
        escreval("      - Opção não disponível, digite novamente!")
        escreval
        escreva("  * Deseja continuar com a compra? (S/N) -> ")
        leia (opc)
        opc <- maiusc(opc)
```

```
se (opc = "N") ou (opc = "NAO") entao
```

```
    interrompa
```

```
fimse
```

```
escreval
escreva("  * Informe o código novamente -> ")
leia(cod)
escreval
```

```
fimenquanto
```

fimse

se (opc = "S") ou (opc = "SIM") entao

escreva(" * Muito bem! Deseja comprar algo mais? (S/N) -> ")

leia(opc)

opc <- maiusc(opc)

fimse

escolha (cod)

caso 1

produtos <- produtos + "Tênis - R\$ 249,99 | "

valor <- (valor + 249.99)

caso 2

produtos <- produtos + "Sandália - R\$ 95,20 | "

valor <- (valor + 95.20)

caso 3

produtos <- produtos + "Chinelo - R\$ 22,90 | "

valor <- (valor + 22.90)

caso 4

produtos <- produtos + "Sapatilha - R\$ 87,10 | "

valor <- (valor + 87.10)

caso 5

produtos <- produtos + "Bota - R\$ 173,90 | "

valor <- (valor + 173.90)

fimescolha

```
ate (opc = "N") ou (opc = "NAO")
```

```
se (produtos = "") entao
```

```
    produtos <- "Nenhum produto foi adicionado! |"
```

```
fimse
```

```
escreval
```

```
escreval(" «-----»")
```

```
escreval("  « RESUMO DO PEDIDO »")
```

```
escreval
```

```
escreval(" - ITENS : | ", produtos)
```

```
escreval(" - TOTAL DO PEDIDO : R$", valor :6:2)
```

```
escreval(" «-----»")
```

Fimalgoritmo