

## Trabalho individual 6

### Objectivos

Aplicar os conceitos aprendidos na cadeira na resolução de exercícios. Os conceitos em referência são sobre Algoritmos Recursivos, dados literais, Funções, Procedimentos e Vetores.

### Instruções:

- **Prazo:** data da próxima aula
- Submissão via e-mail: <gquissico@gmail.com>;
- Assunto do e-mail: USTM\_EDA\_<suaTurma>\_<seuNome>\_Trabalho Individual 2
- Corpo de e-mail: Escreva alguma coisa por exemplo Saudações | Eis o trabalho sobre .....
- Nome do ficheiro: USTM\_EDA\_<suaTurma>\_<seuNome>\_Trabalho Individual 2.pdf

Apresente suas dúvidas nas aulas (ou nas outras alternativas de comunicação)

1. Escreva um procedimento que receba 3 notas de um aluno por parâmetro e uma letra. Se a letra for “A” o procedimento calcula a média aritmética das notas do aluno, se for “P”, a sua média ponderada (pesos: 5, 3 e 2) e se for H, a sua média harmónica. A média calculada também deve ser retornada por parâmetro.
2. Desenvolva uma função que receba por parâmetro um valor inteiro positivo, e retorne o valor lógico “Verdadeiro” se o número recebido for primo e “Falso” no caso contrário.
3. Faça uma função que verifica se um valor recebido por parâmetro é ou não perfeito (um número é perfeito quando ele é igual a soma dos seus divisores exceptuando ele).
4. Desenvolva uma função que simula uma máquina de calcular que recebe do utilizador uma sequência de números e operadores e retorne o resultado da expressão (tenha em atenção as prioridades dos operadores).
5. Dado um vetor **V** de **n** elementos inteiros, desenvolva uma função recursiva para:
  - a. Encontrar o valor máximo do vetor
  - b. Encontrar o valor mínimo do vetor
  - c. Encontrar a soma dos elementos do vetor
6. Desenvolva uma função ou procedimento recursivo para:
  - a. Escrever os números de **n** até 0
  - b. Escrever os números de 0 até **n**
  - c. Calcular a soma de todos os números inteiros entre 1 e **n**
  - d. Calcular o produto de todos os números inteiros entre 1 e **n**
  - e. Calcular o produto de todos os quadrados dos números inteiros entre 1 e **n**
7. Desenvolva um programa que recebe do utilizador uma palavra e apresente na saída a mesma palavra mas na ordem reversa.
8. Desenvolva um programa que receba uma lista de nomes de estudantes para um vetor e um caractere, e apresente quais dos nomes armazenados apresentam aquele caractere e quantas vezes ele aparece em cada nome.
9. Desenvolva uma função que recebe por parâmetro dois vetores de inteiros com tamanho 10 e retorne um vetor que será a união dos dois anteriores.
10. Faça uma função que receba 50 valores reais e retorne um vetor com os mesmos valores em ordem crescente.

11. Desenvolva um procedimento que imprima os 10 primeiros números primos acima do número fornecido pelo utilizador.
12. Escreva um programa que receba para um vetor uma lista de nome de estudantes e imprima a mesma lista em ordem alfabética.
13. Desenvolva uma função recursiva para calcular o fatorial de um número, sabendo que  $4! = 2 \times 3 \times 4$ .
14. Faça um procedimento que receba, por parâmetro, um vetor  $k(15)$  de inteiros e retorne também por parâmetro um vetor  $p$  que conterá apenas os valores primos da lista de valores em  $k$ .

Algoritmo "1. Media Aritmetica, Ponderada e Harmonica"

// Autor(a) : Edilson Alexandre Cuamba

// Data atual : 6/4/2021

Var

// Seção de Declarações das variáveis

notas: vetor [1..3] de real

procedimento start()

var

contador:inteiro

inicio

para contador de 1 ate 3 faca

notas[contador] <- pedeValor(contador)

fimpara

queMedia()

fimprocedimento

procedimento queMedia()

var

verificacao:logico

forma:caracter

inicio

repita

verificacao <- verdadeiro

Escreval("Deseja calcular como a media?")

Escreval("A - Media Aritimetica")

Escreval("P - Media Ponderada")

Escreval("H - Media Harmonica")

Leia(forma)

escolha forma

caso "A"

mediaAritmetica()

caso "P"

mediaPonderada()

caso "H"

mediaHarmonica()

outrocaso

verificacao <- falso

fimescolha

ate verificacao

fimprocedimento

funcao pedeValor(numero:inteiro):real

var

nota:real

inicio

repita

Escreva("Digita a nota do ")

Escreva(numero)

Escreval(" aluno: (Não digitar nada equivale a 0)")

Leia(nota)

ate nota >= 0

retorne nota

fimfuncao

```
procedimento mediaAritmetica()
var
    media:real
inicio
    media <- (notas[1]+notas[2]+notas[3])/3;
    Escreva("A media Aritmetica é ")
    Escreva(media)
fimprocedimento
```

```
procedimento mediaPonderada()
var
    media:real
inicio
    media <- ((notas[1]*5)+(notas[2]*3)+(notas[3]*2))/(5+3+2)
    Escreva("A media Ponderada é ")
    Escreva(media)
fimprocedimento
```

```
procedimento mediaHarmonica()
var
    media:real
inicio
    media <- 3/((1/notas[1]) + (1/notas[2]) + (1/notas[3]))
    Escreva("A media Harmonica é ")
    Escreva(media)
fimprocedimento
```

```
Inicio
    // Seção de Comandos, procedimento, funções, operadores, etc...
    start()
```

Fimalgoritmo

Algoritmo "2. verificar se é numero primo"

// Autor(a) : Edilson Alexandre Cuamba

// Data atual : 6/7/2021

Var

valor, divisor: inteiro

procedimento start()

var

resultado: logico

inicio

divisor <- 2

valor <- pedirValor()

resultado <- verificaSeEhPrimo(valor)

Escreval()

se resultado entao

Escreval("V")

Escreval("O valor " + numpcarac(valor) + " é Primo")

senao

Escreval("F")

Escreval("O valor " + numpcarac(valor) + " não é Primo")

fimse

fimprocedimento

funcao pedirValor():inteiro

var

numero, divisor: inteiro

inicio

Escreval("Resultado V significa que é primo, F que não é primo.")

Escreva("Por favor digite uma valor para verificar se é primo: ")

Leia(numero)

retorne numero

fimfuncao

//Um valor só é primo se for divisiver por 1 e por se mesmo e caso seja maior

//que 9 será primo se a divisão por valores de 2 até 9 terem o resto diferente

//de zero (0)

funcao verificaSeEhPrimo(numero:inteiro):logico

var

inicio

se divisor <> numero entao

se(numero MOD divisor) <> 0 entao

divisor <- divisor + 1

se divisor > 9 entao

retorne verdadeiro

senao

retorne verificaSeEhPrimo(numero)

fimse

senao

retorne falso

fimse

senao

divisor <- divisor + 1

retorne verificaSeEhPrimo(numero)

fimse

fimfuncao

Inicio

// Seção de Comandos, procedimento, funções, operadores, etc...

start()

Fimalgoritmo

### Algoritmo "3. VERIFICAR SE UM NUMERO É PERFEITO"

// Autor(a) : Edilson Alexnadre Cuamba

// Data atual : 6/7/2021

```
Var
    // Seção de Declarações das variáveis
    divisorActual:inteiro
    somatorio:inteiro
    numero: inteiro

procedimento start()
var
    resultado: logico
inicio
    numero <- pedirValor()
    resultado <- verificaPerfeicao(numero)
    imprimirResultado(resultado)
fimprocedimento

funcao pedirValor():inteiro
var
    numero: inteiro
inicio
    Escreval("Por favor digite um valor seja ele inteiro ou inteiro")
    Escreva("para verificar se é perfeito: ")
    Leia(numero)
    retorne numero
fimfuncao

funcao verificaPerfeicao(numero:inteiro):logico
var
inicio
    divisorActual <- divisorActual+1
    se divisorActual <> numero entao
        se (numero MOD divisorActual) = 0 entao
            somatorio <- somatorio + divisorActual
            retorne verificaPerfeicao(numero)
        senao
            retorne verificaPerfeicao(numero)
        fimse
    senao
        se somatorio = numero entao
            retorne verdadeiro
        senao
            retorne falso
        fimse
    fimse
fimfuncao

procedimento imprimirResultado(resultado:logico)
var
inicio
    se resultado entao
        Escreval("O valor " + numpcarac(numero) + " é perfeito!")
    senao
        Escreval("O valor " + numpcarac(numero) + " não é perfeito!")
    fimse
fimprocedimento

Inicio
    // Seção de Comandos, procedimento, funções, operadores, etc...
    start()
Fimalgoritmo
```

```
Algoritmo "4. CALCULADORA"
// Autor(a) : Edilson Alexandre Cuamba
// Data atual : 6/7/2021
Var
    valores: vetor [1..3] de caracter
```

```
procedimento start()
var
    valor_1, valor_2: real
inicio
    input()
    valor_1 <- caracpnum(valores[1])
    valor_2 <- caracpnum(valores[2])

    escolha valores[3]
    caso "+"
        soma(valor_1, valor_2)
    caso "-"
        subtracao(valor_1, valor_2)
    caso "*"
        multiplicacao(valor_1, valor_2)
    caso "/"
        divisao(valor_1, valor_2)
    outrocaso
        Escreva("O sinal digitado é invalido!")
    fimescolha
fimprocedimento
```

```
procedimento input()
var
    //valores: vetor [1..3] de caracter
    contador: inteiro
inicio
    para contador de 1 ate 3 faca
        se contador < 3 entao
            Escreva("Insira o "+ numpcarac(contador) +" valor: ")
            Leia(valores[contador])
        senao
            Escreva(numpcarac(contador) + " - Insira o sinal da operação '+, -, /, *' :)")
            Leia(valores[contador])
        fimse
    fimpara
fimprocedimento
```

```
procedimento soma(valor_1, valor_2:real)
var
inicio
    Escreval("O resultado da soma entre " + numpcarac(valor_1) + " e " + numpcarac(valor_2) + " é " +
numpcarac(valor_1 + valor_2))
fimprocedimento
```

```
procedimento subtracao(valor_1, valor_2:real)
var
inicio
    Escreval("O resultado da subtração entre " + numpcarac(valor_1) + " e " + numpcarac(valor_2) + " é " +
numpcarac(valor_1 - valor_2))
fimprocedimento
```

```
procedimento multiplicacao(valor_1, valor_2:real)
```

```
var
```

```
inicio
```

```
    Escreval("O resultado da multiplicação entre " + numpcarac(valor_1) + " e " + numpcarac(valor_2) + " é " +  
numpcarac(valor_1 * valor_2))
```

```
fimprocedimento
```

```
procedimento divisao(valor_1, valor_2:real)
```

```
var
```

```
inicio
```

```
    Escreval("O resultado da divisão entre " + numpcarac(valor_1) + " e " + numpcarac(valor_2) + " é " +  
numpcarac(valor_1 / valor_2))
```

```
fimprocedimento
```

```
Inicio
```

```
    start()
```

```
Fimalgoritmo
```



# Algoritmo "5. MAIOR, MENOR E SOMA DE VALORES DE UM VETOR"

// Autor(a) : Edilson Alexandre Cuamba

// Data atual : 6/7/2021

Var

valores: vetor [1..5] de inteiro

maior, menor, somatorio: inteiro

posicao: inteiro

procedimento start()

var

inicio

para posicao de 1 ate 5 faca

Escreval("Digite o valor " + numpcarac(posicao))

Leia(valores[posicao])

fimpara

posicao<-1

menor<- valores[1]

processamento()

fimprocedimento

procedimento processamento()

var

inicio

se posicao > 5 entao

Escreval("O maior numero é " + numpcarac(maior))

Escreval("O menor numero é " + numpcarac(menor))

Escreval("O somatorio é " + numpcarac(somatorio))

senao

somatorio<- somatorio + valores[posicao]

se valores[posicao] > maior entao

maior <- valores[posicao]

fimse

se (valores[posicao] < menor) entao

menor <- valores[posicao]

fimse

posicao<- posicao + 1

processamento()

fimse

fimprocedimento

Inicio

start()

Fimalgoritmo

Algoritmo "6. UMA FUNCAO PARA A,B,C,D,E"

// Autor(a) : Edilson Alexandre Cuamba

// Data atual : 6/7/2021

Var

// Seção de Declarações das variáveis

valores: vetor [1..5] de inteiro

comprimento, contador, somatorio, posicao, produto: inteiro

procedimento start()

var

res, res\_produto: inteiro

inicio

comprimento<-5

para contador de 1 ate comprimento faca

Escreva("Escreva o valor do " + numpcarac(contador) + " elemento: ")

Leia(valores[contador])

fimpara

escreve\_de\_zero\_ate\_n(1)

escreval()

escreve\_de\_n\_ate\_zero(1)

escreval()

res <- escreve\_a\_soma\_de\_numeros\_inteiros(0)

escreval("O resultado do somatorio dos inteiro é " + numpcarac(res))

res\_produto <- escreve\_a\_produto\_de\_numeros\_inteiros(0)

escreval("O resultado do produto dos inteiro é " + numpcarac(res\_produto))

res\_produto <- escreve\_a\_produto\_do\_quadrado\_dos\_numeros\_inteiros(0)

escreval("O resultado do produto do quadrado dos inteiros é " + numpcarac(res\_produto))

fimprocedimento

procedimento escreve\_de\_n\_ate\_zero(posicao: inteiro)

var

varloreTemporario: inteiro

inicio

se posicao = comprimento entao

para contador de 1 ate comprimento faca

Escreva(numpcarac(valores[contador]) + " - ")

fimpara

senao

para contador de posicao ate comprimento faca

se valores[posicao] < valores[contador] entao

varloreTemporario <- valores[posicao]

valores[posicao]<- valores[contador]

valores[contador]<- varloreTemporario

fimse

fimpara

posicao<- posicao+1

escreve\_de\_n\_ate\_zero(posicao)

fimse

fimprocedimento

```

funcao escreve_a_soma_de_numeros_inteiros(posicao:inteiro):inteiro //Retornamos a soma total dos valores
do vetor
var
inicio
//Incrementamos uma unidade na nossa variavel de escape
posicao <- posicao + 1
se posicao = comprimento entao
//Retornamos o ultimo valor do array/vetor valores[limite_superior]
retorne valores[posicao]
senao
//Somamos o valor retornado e o valor da posicao actual
produto <- escreve_a_soma_de_numeros_inteiros(posicao) + valores[posicao]
//retornamos o resultado da soma para voltar a somar e retornar
retorne produto
fimse
fimfuncao

```

```

funcao escreve_a_produto_de_numeros_inteiros(posicao:inteiro):inteiro
var
inicio
posicao <- posicao + 1
se posicao = comprimento entao
retorne valores[posicao]
senao
somatorio <- escreve_a_produto_de_numeros_inteiros(posicao) * valores[posicao]
retorne somatorio
fimse
fimfuncao

```

```

funcao escreve_a_produto_do_quadrado_dos_numeros_inteiros(posicao:inteiro):inteiro
var
inicio
posicao <- posicao + 1
se posicao = comprimento entao
retorne Quad(valores[posicao])
senao
somatorio <- escreve_a_produto_do_quadrado_dos_numeros_inteiros(posicao) * Quad(valores[posicao])
retorne somatorio
fimse
fimfuncao

```

```

procedimento escreve_de_zero_ate_n(posicao:inteiro)
var
varloreTemporario:inteiro
inicio
se posicao = comprimento entao
para contador de 1 ate comprimento faca
Escreva(numpcarac(valores[contador]) + " - ")
fimpara
senao
para contador de posicao ate comprimento faca
se valores[posicao] > valores[contador] entao
varloreTemporario <- valores[posicao]
valores[posicao]<- valores[contador]
valores[contador]<- varloreTemporario
fimse
fimpara
posicao<- posicao+1
escreve_de_zero_ate_n(posicao)
fimse
fimprocedimento
Inicio start() Fimalgoritmo

```

Algoritmo "7. RECEBE E IMPRIME PALAVRA INVERTIDA"

// Autor(a) : Edilson Alexandre Cuamba

// Data atual : 06/08/21

Var

// Seção de Declarações das variáveis

palavra:caracter

contador: inteiro

procedimento start()

var

inicio

Escreva("Digite a palavra que deseja inverter: ")

Leia(palavra)

invertePalavra(palavra)

fimprocedimento

procedimento invertePalavra(palavra:caracter)

var

posicao:inteiro

inicio

posicao <- compr(palavra)

para contador de 1 ate compr(palavra) faca

Escreva(copia(palavra, posicao, 1))

posicao <- posicao - 1

fimpara

fimprocedimento

Inicio

// Seção de Comandos, procedimento, funções, operadores, etc...

start()

Fimalgoritmo

Algoritmo "7. RECEBE E IMPRIME PALAVRA INVERTIDA"

// Autor(a) : Edilson Alexandre Cuamba

// Data atual : 06/08/21

Var

// Seção de Declarações das variáveis

palavra:caracter

contador: inteiro

procedimento start()

var

inicio

Escreva("Digite a palavra que deseja inverter: ")

Leia(palavra)

invertePalavra(palavra)

fimprocedimento

procedimento invertePalavra(palavra:caracter)

var

posicao:inteiro

inicio

posicao <- compr(palavra)

para contador de 1 ate compr(palavra) faca

Escreva(copia(palavra, posicao, 1))

posicao <- posicao - 1

fimpara

fimprocedimento

Inicio

// Seção de Comandos, procedimento, funções, operadores, etc...

start()

Fimalgoritmo