

Programação I - 2022/1

INF01106

Profa. Dra. Annabell D.R. Tamariz

Universidade Estadual do Norte Fluminense Darcy Ribeiro

26 de abril de 2022

Algoritmos

Estruturação de Programas

Estruturas de Controle de Fluxo

- Estruturas Condicionais

- Estruturas de Repetição

Modularização

- Escopo das Variáveis - Modularização

Exercícios

- Exercícios de Fixação

- Exercícios para casa

Algoritmos

Estruturação de Programas

Estruturas de Controle de Fluxo

- Estruturas Condicionais

- Estruturas de Repetição

Modularização

- Escopo das Variáveis - Modularização

Exercícios

- Exercícios de Fixação

- Exercícios para casa

Exercícios para praticar os conceitos

1. Faça um algoritmo em pseudocódigo para escrever os números pares de 0 a 100.
2. Faça um algoritmo em pseudocódigo para escrever a série de Fibonacci = (0, 1, 1, 2, 3, 5, 8, 13, 21, 34, ...) enquanto o termo a ser impresso for menor que 300.
3. Escreva um algoritmo em pseudocódigo que requisita dois números e faz a soma deles e depois pergunta se o usuário quer fazer o cálculo novamente.
4. Escreva um Pseudocódigo que recebe um número e conta a partir deste número até 100.
5. Escreva um Pseudocódigo para ler 10 números e dizer se cada um é: nulo, positivo ou negativo.

Respostas dos Exercícios

1- Faça um algoritmo em pseudocódigo para escrever os números pares de 0 a 100.

```
algoritmo "par-0-a-100"  
var par: inteiro  
Inicio  
para par de 0 ate 100 faca  
se( $\text{par} \% 2 = 0$ ) entao  
escreval (par)  
fimse  
fimpara  
fimalgoritmo
```

Respostas dos Exercícios

2- Faça um algoritmo em pseudocódigo para escrever a série de Fibonacci = (0, 1, 1, 2, 3, 5, 8, 13, 21, 34, ...) enquanto o termo a ser impresso for menor que 300.

algoritmo "fibonacci"

var proximo, atual, anterior: inteiro

Inicio

proximo:= 0 atual:= 0 anterior:= 1

se proximo = 0 entao

 escreval (proximo)

fimse

enquanto (proximo<300) faca

 escreval (proximo)

 proximo:= (atual + anterior)

 anterior:= atual

 atual:= proximo

fimenquanto

fimalgoritmo

Respostas dos Exercícios

3- Pseudocódigo que requisita dois números e faz a soma deles e depois pergunta se o usuário quer fazer o cálculo novamente.

algoritmo “repete soma”

var n1, n2, soma: real resp: caracter

Inicio

repita

escreval ("Digite dois numeros para serem somados:") leia (n1,n2)

soma:= $n1+n2$

escreval ("A soma eh:",soma)

escreval ("Digite algo p/ fazer novo calculo e fim p/ encerrar") leia (resp)

ate (resp = "fim")

fimalgoritmo

Respostas dos Exercícios

4- Escreva um Pseudocódigo que recebe um número e conta a partir deste número até 100.

```
algoritmo "Conta ate 100"
```

```
var  a: inteiro
```

```
inicio
```

```
escreval("Digite um numero inteiro menor que  
100")  leia(a)
```

```
repita
```

```
    a<-a+1
```

```
    escreval (a)
```

```
ate (a=100)
```

```
fimalgoritmo
```


Respostas dos Exercícios

5- Escreva um Pseudocódigo para ler 10 números e dizer se cada um é: nulo, positivo ou negativo.

algoritmo "definir numero"

var n1: inteiro

Inicio

para n1 de 0 ate 10 faca

 escreval ("Digite um numero:") leia (n1)

 se (n1=0) entao

 escreval ("nulo")

 fimse

 se (n1<0) entao

 escreval ("Numero negativo")

 fimse

 se (n1>0) entao

 escreval ("Numero positivo")

 fimse

fimpara

Exemplos em Linguagens

Pascal:

```
read(x);  
while x <> 0 do begin  
  writeln(x);  
  repeat  
    read(next);  
  until next <> x;  
  x := next;  
end;
```

Exemplos em Pseudocódigo

```
algoritmo "exerc-fat"  
var  
i,num,fat: inteiro  
inicio  
  repita  
    escreva ("Digite um número natural: ")  
    leia (num)  
    ate (num>=0)  
    fat <- 1  
    para i de 2 ate num faca  
      fat <- fat * i  
    fimpara  
    escreva ("O fatorial de ", num, "é: ", fat)  
  fimalgoritmo
```

algoritmo "exerc-fat2"

var

num, aux: inteiro

inicio

 repita

 escreva ("Digite um número natural: ")

 leia (num)

 ate (num \geq 0)

 se (num=0) entao

 escreva ("O fatorial é: 1")

 senao

 para aux de num-1 ate 2 passo -1 faca

 num <- num * aux

 fimpara

 escreva ("O fatorial é: ", num)

 fimse

fimalgoritmo

Exercícios para praticar os conceitos

1. Escreva um algoritmo (pseudocódigo) para calcular o valor da série, para 5 termos.

$$S = -\frac{1}{2!} + \frac{2}{4!} - \frac{3}{6!} + \dots$$

Respostas aos exercícios

```
algoritmo "exerc-"  
var  
ind, i, fat: inteiro  
s: real  
inicio  
  s <- 0  
  para ind de 1 ate 5 faca  
    fat <- 1  
    para i de 2 ate ind*2 faca  
      fat <- fat * i  
    fimpara  
  segue no próximo slide....  
finalgoritmo
```

Respostas aos exercícios

inicio

continuação....

para ind de 1 ate 5 faca

fat <- 1

para i de 2 ate ind*2 faca

fat <- fat * i

fimpara

se (ind%2=0) entao

s <- s + ind / fat

senao

s <- s - ind / fat

fimse

fimpara

escreval ("O valor do somatório é: ", s)

fimalgoritmo

Exercícios para Casa...

1. Escreva um Pseudocódigo que calcula o valor do imposto de renda de uma pessoa física, com as seguintes condições: se o salário ≥ 3.000 , alíquota será 15%. Se $3.000 > \text{salário} \geq 1500$, alíquota será 7%. Se salário < 1500 , isento.
2. Escreva um algoritmo em pseudocódigo que calcule $N!$, sendo que N é um inteiro fornecido pelo usuário e que $0! = 1$, por definição.
3. Elabore um algoritmo em pseudocódigo para cada estrutura de repetição (enquanto, repita e para) imprimir a tabuada do número 5.

Exercícios para Casa...

1. Escreva um algoritmo em pseudocódigo para calcular o valor da série, para N termos. Onde o valor de N será fornecido pelo usuário.

$$S = -\frac{1}{2!} + \frac{2}{4!} - \frac{3}{6!} + \dots$$

Exercícios para Casa... Modularização

1. Construa um algoritmo em pseudocódigo que tenha a capacidade de efetuar uma multiplicação entre valores naturais quaisquer e a capacidade de calcular o fatorial de um número natural qualquer.
 - ▶ Tanto no cálculo da multiplicação quanto no cálculo do fatorial os únicos operadores aritméticos que podem ser utilizados são os de soma e subtração.
 - ▶ O algoritmo em questão deve possibilitar ao usuário fazer a seleção de qual operação será realizada.
 - ▶ As entradas devem ser validadas e o conceito de modularização deve ser aplicado.