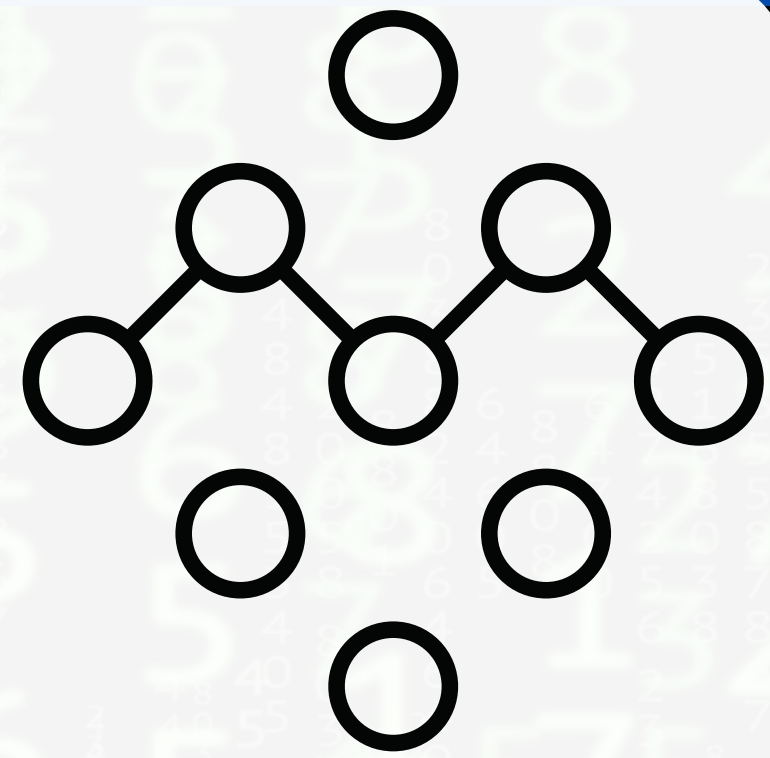


Curso de

Lógica de Programação

Por

Jeiverson C. V. M. Santos

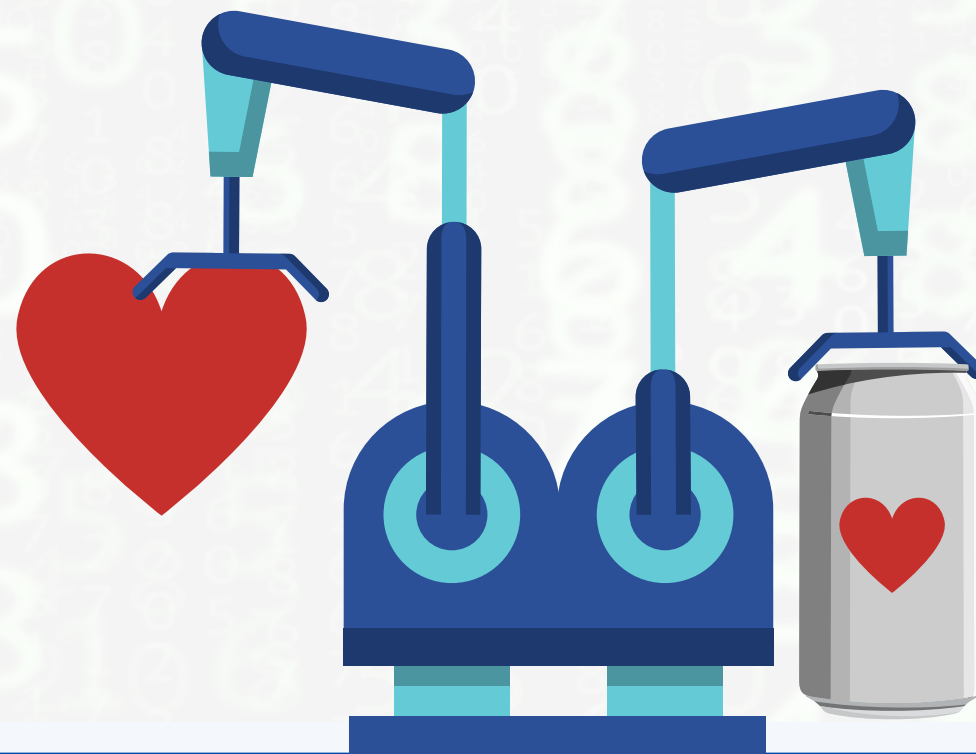


Aula 14

Funções

*"O amor verdadeiro é como uma função matemática.
Você dá algo, em função recebe algo em troca."*

Ruizuma



Primeiro

Vamos ver o resultado do "para casa"



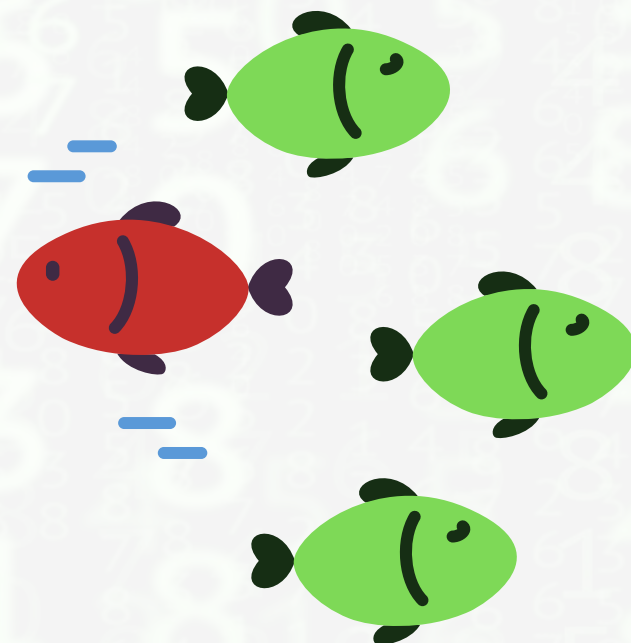
Funções | Procedimentos

Rotina ✓

Pode retornar
um resultado ✓

Rotina ✓

Não pode retornar
um resultado ✗



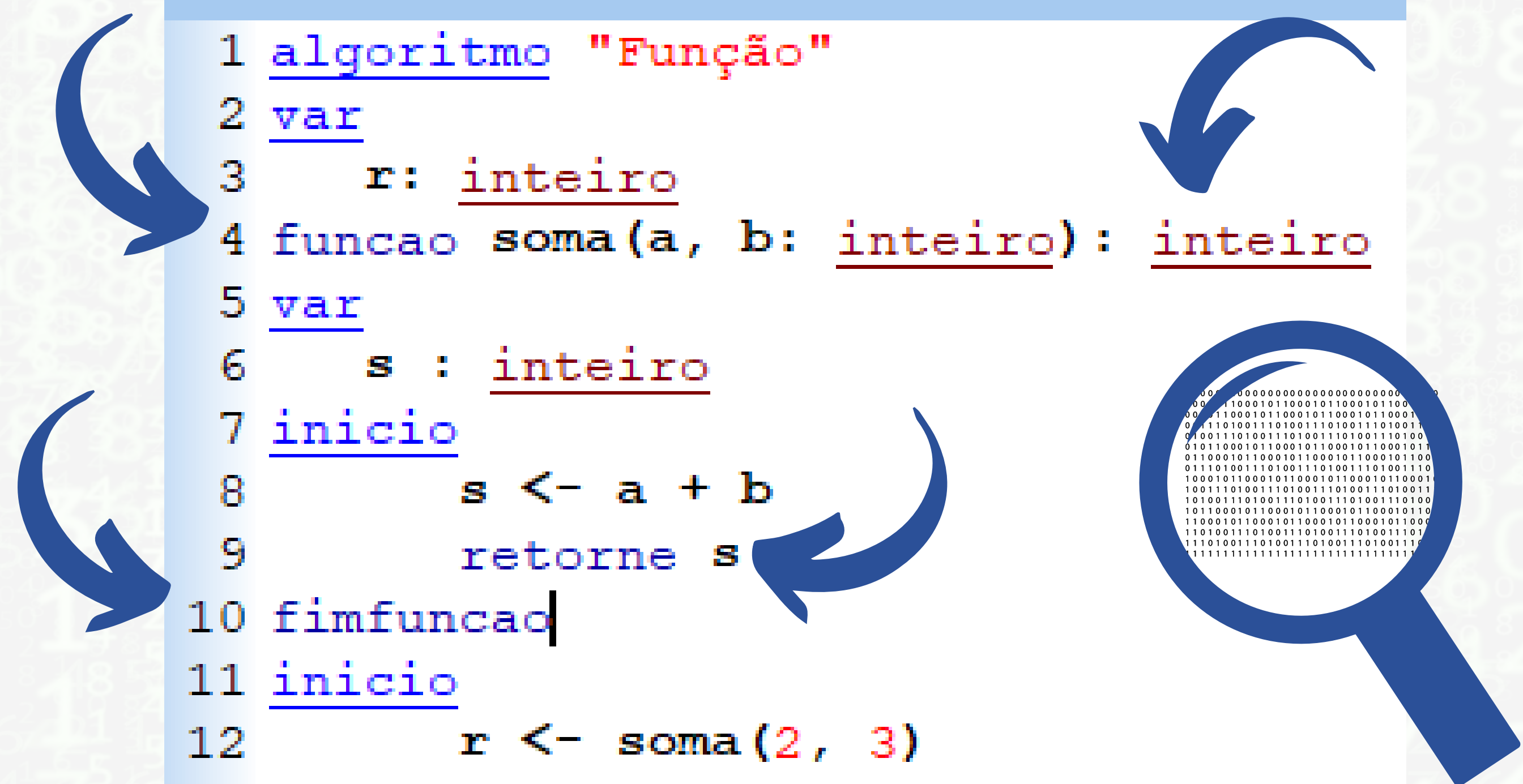
Exemplo

```
1 algoritmo "Procedimento"  
2 var  
3 procedimento soma(a, b: inteiro)  
4 var  
5     s : inteiro  
6 inicio  
7     s <- a + b  
8     escreva(s)  
9 fimprocedimento  
10 inicio  
11     soma(2, 3)  
12 fimalgoritmo |
```

```
1 algoritmo "Função"  
2 var  
3     r: inteiro  
4 funcao soma(a, b: inteiro): inteiro  
5 var  
6     s : inteiro  
7 inicio  
8     s <- a + b  
9     retorne s  
10 fimfuncao  
11 inicio  
12     r <- soma(2, 3)  
13     escreva(r)  
14 fimalgoritmo
```



Olhando mais de perto...



The diagram illustrates a code analysis process. Four blue curved arrows point from the left towards specific lines of code: line 1, line 4, line 9, and line 11. A large blue magnifying glass is positioned over the code, with its lens centered on line 9, which contains the statement 'retorne s'. The background of the slide features a faint, repeating pattern of binary digits (0s and 1s).

```
1 algoritmo "Função"  
2 var  
3   r: inteiro  
4 funcao soma(a, b: inteiro): inteiro  
5 var  
6   s : inteiro  
7 inicio  
8   s <- a + b  
9   retorne s  
10 fimfuncao  
11 inicio  
12   r <- soma(2, 3)  
13   escreva(r)  
14 fimalgoritmo
```

Lembram?

Funções Aritméticas

Abs	Valor Absoluto	Abs (-10)	10
Exp	Exponenciação	Exp (3,2)	9
Int	Valor Inteiro	Int (2.9)	2
RaizQ	Raiz Quadrada	RaizQ (25)	5
Pi	Retorna Pi	Pi	3.14...
Sen	Seno (rad)	Sen (0.523)	0.5
Cos	Cosseno (rad)	Cos (3.14)	-1
Tan	Tangente (rad)	Tan (0.785)	1
GraupRad	Graus para Radiano	GraupRad (30)	0.523

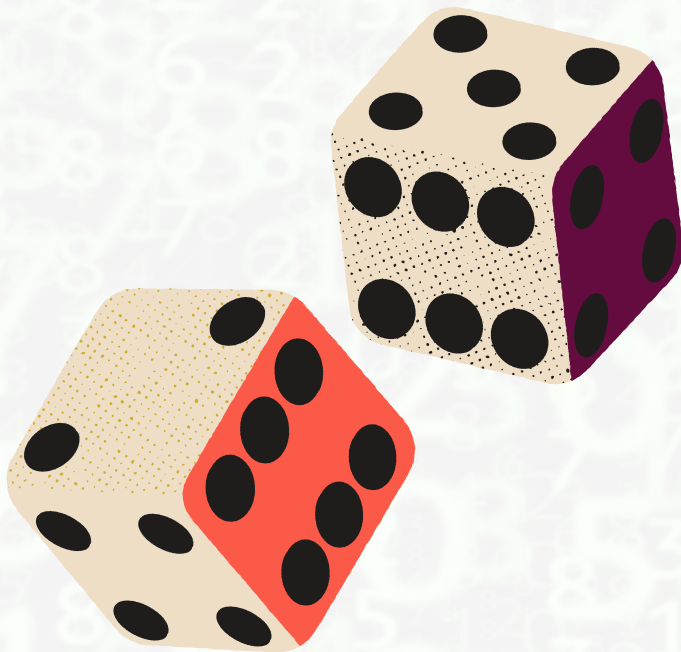
Exercício

Fazer um algoritmo com uma função que retorne a palavra "PAR" ou "ÍMPAR", dependendo do número que for digitado.



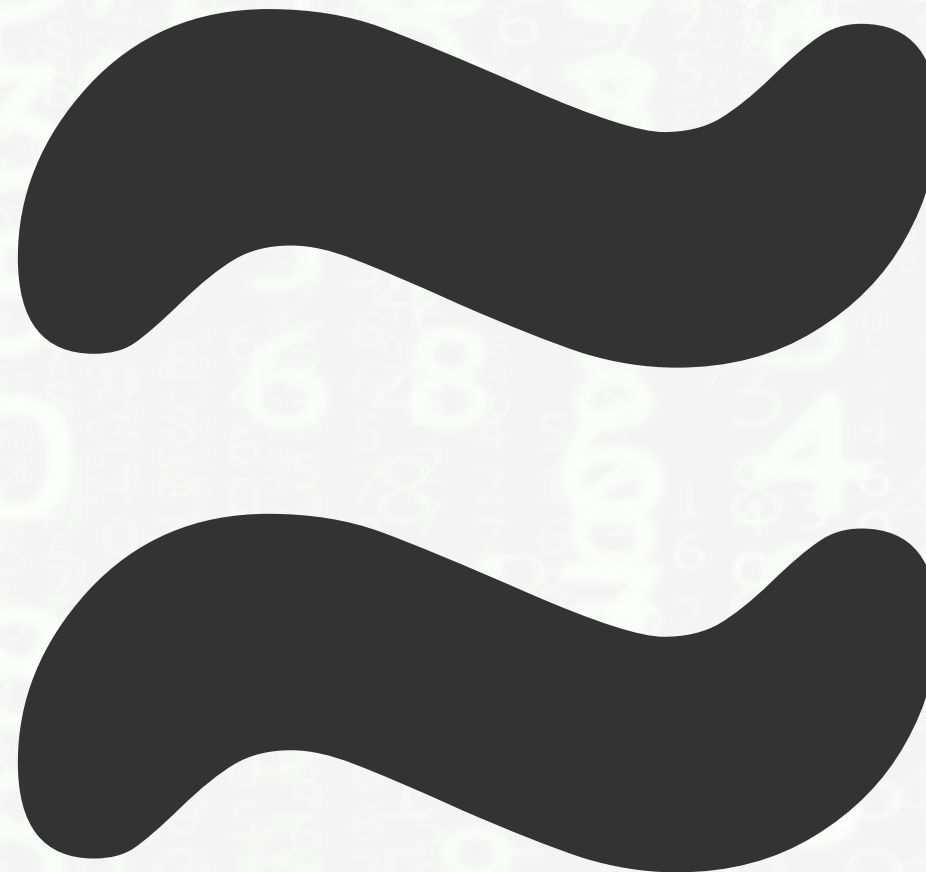
Resposta

```
1 algoritmo "Par | Impar"
2 var
3     n: inteiro
4     r: caractere
5 funcao ParImpar(n: inteiro) : caractere
6 inicio
7     se n % 2 = 0 entao
8         retorne "PAR"
9     senao
10        retorne "ÍMPAR"
11    fimse
12 fimfuncao
13 inicio
14     escreva("Digite um número: ")
15     leia(n)
16     r <- ParImpar(n)
17     escreva(r)|
18 fimalgoritmo
```



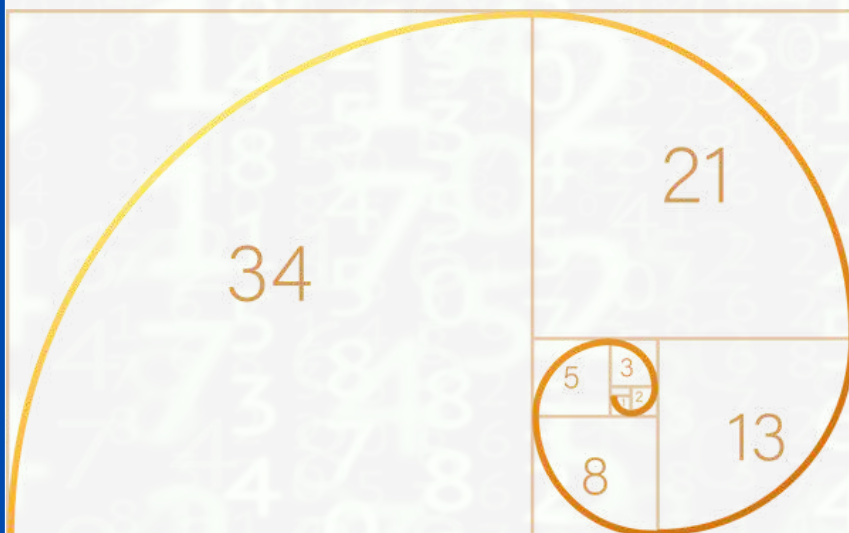
Passagem por valor x referência

Mesma coisa do procedimento



Exemplo

```
1 algoritmo "Fibonacci"
2 var
3   i, f1, f2, f3: inteiro
4 funcao ProximoFib(var n1, n2: inteiro): inteiro
5 var
6   n3: inteiro
7 inicio
8   n3 <- n1 + n2
9   n1 <- n2
10  n2 <- n3
11  retorne n3
12 fimfuncao
13 inicio
14   f1 <- 1
15   f2 <- 1
16   escreva(f1, f2)
17   para i <- 3 ate 10 passo 1 faca
18     f3 <- ProximoFib(f1, f2)
19     escreva(f3)
20   fimpara
21 fimalgoritmo |
```



Outras funções prontas

nome

"Jeiverson"

compr(nome) → 9

copia(nome,7,3) → "son"

maiusc(nome) → "JEIVERSON"

minusc(nome) → "jeiverson"

pos("son",nome) → 7



tabela ASCII

asc("J") → 74

carac(74) → "J"

Exemplo

```
1 algoritmo "Analizador de Nomes"
2 var
3     n: caractere
4 inicio
5     escreva("Digite seu nome: ")
6     leia(n)
7     escreval("Total de letras: ", compr(n))
8     escreval("Tudo maiúsculo: ", maiusc(n))
9     escreval("Tudo minúsculo: ", minusc(n))
10    escreval("Primeira letra: ", copia(n,1,1))
11    escreval("Última letra: ", copia(n,compr(n),1))
12    escreval("Posição da letra J: ", pos("N", maiusc(n)))
13 fimalgoritmo |
```


Para casa

Faça um algoritmo que escreva o nome do usuário ao contrário.

