



**INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
RIO GRANDE DO NORTE**

Algoritmos e Técnicas de Programação

Procedimentos e Funções

filipe.raulino@ifrn.edu.br

Estrutura de Um algoritmo

Nome do Programa

Inclusão de Bibliotecas

Declaração de Variáveis Globais

Criação de Procedimentos e Funções

Indicador de Início do Programa (Início)

Códigos do Programa

Indicador de Término do Programa (Fim)

Procedimentos

- ▶ São subrotinas (módulos ou métodos) de programas, capazes de executar uma tarefa definida pelo programador, mas que não retorna nenhum valor. Os programas desenvolvidos com procedimentos são ditos modulares.
- ▶ As principais vantagens em criar programas usando subrotinas são:
 - ▶ Melhor organização do programa;
 - ▶ Reutilização da subrotina em outras partes do programa; e
 - ▶ Facilidade de manutenção do código.

Estrutura de um Procedimento

- ▶ Todo procedimento deve ter um identificador;
- ▶ Pode possuir um conjunto de parâmetros;
- ▶ Permite declaração de variáveis locais; e
- ▶ Possui um bloco de instruções.

```
nome_procedimento (parâmetros)
variáveis locais
início
    instruções . . .
fim.
```

Tipos de Passagem de Parâmetros

- ▶ A passagem de parâmetros pode ser de dois tipos:
 - ▶ por valor; e
 - ▶ por referência.

Valor

Informamos o valor a ser trabalhado e indiferente de quais modificações serão feitas com essas informações, seus valores originais permanecem o mesmo.

Referência

Informamos o valor a ser trabalhado e de acordo com as mudanças que vão sofrendo, os valores originais vão sendo atualizados. Passagem por referência somente é feita passando-se variáveis como parâmetro.

Exemplo Procedimento

```
algoritmo "ExemploProcedimento"
```

```
procedimento exibir_cumprimento
```

```
inicio
```

```
    escreva("Oi Mundo!")
```

```
fimprocedimento
```

```
inicio
```

```
    exibir_cumprimento
```

```
fimalgoritmo
```

Exemplo Procedimento

```
algoritmo "ExemploProcedimento"  
  
procedimento exibir_cumprimento(n: caracter)  
inicio  
    escreva("Oi ", n, "!")  
fimprocedimento  
  
var nome: caracter  
  
inicio  
    escreva("Digite seu nome:")  
    leia(nome)  
    exibir_cumprimento(nome)  
finalgoritmo
```

Funções

- ▶ São rotinas similares aos procedimentos, só que retornam um valor após cada chamada. Uma função não deverá simplesmente ser chamada, como no caso dos procedimentos, mas deverá ser atribuída à alguma variável.

Estrutura de uma Função

- ▶ Todo função deve ter um identificador;
- ▶ Pode possuir um conjunto de parâmetros;
- ▶ **Obrigatoriamente deve retornar um valor;**
- ▶ Permite declaração de variáveis locais; e
- ▶ Possui um bloco de instruções.

```
nome_função (parâmetros) : tipo de retorno  
variáveis locais  
início  
    instruções . . .  
fim.
```

Exemplo Função

```
algoritmo "ExemploFuncao"  
  
funcao somar(a,b:real):inteiro  
    var r:inteiro  
    inicio  
        r <- a + b  
    retorne r  
fimfuncao  
  
var valor1, valor2, resultado :inteiro  
  
inicio  
    escreva("Digite o primeiro valor:")  
    leia(valor1)  
    escreva("Digite o segundo valor:")  
    leia(valor2)  
    resultado <- somar(valor1, valor2)  
    escreva("A soma é ", resultado)  
fimalgoritmo
```

Exercício

http://10.68.2.42/al_exercicio_10.pdf

http://10.68.2.42/al_exercicio_11.pdf