Funções e Procedimentos

ECT3201 - Linguagem de Programação

Prof. Éverton Santi

Motivação

Programas maiores exigem organização.

Com funções e procedimentos, podemos:

- dividir o problema em partes menores
- reutilizar código
- facilitar manutenção e testes

Módulos

Em termos gerais:

Entrada -> Processamento -> Saída

O processamento pode ser decomposto em módulos.

Exemplo 1

Problema: sistema de notas de uma turma.

Possíveis módulos:

- ler nota válida
- calcular média
- classificar situação
- mostrar relatório final

Função e Procedimento

Função

- processa e **retorna** valor

Procedimento

- processa e **não retorna** valor (`void`)

Exemplo 2

Crie:

- uma função para somar dois números reais
- um procedimento para mostrar uma linha de separação na tela

Considere linha de separação como uma sequência de caracteres, por exemplo:

```
------------------------------
```

Estrutura de Função

```
tipo_retorno nome(parametros) {
    // processamento
    return valor;
}
```

Tipos de retorno comuns:

- `int`, `double`, `bool`

Estrutura de Procedimento

```
void nome(parametros) {
    // processamento
}
```

Em C++, procedimentos são funções com retorno `void`.

Parâmetros e Argumentos

Parâmetros:

- variáveis formais da função

Argumentos:

- valores informados na chamada

Exemplo 3

Crie uma função que:

- receba peso e altura
- calcule o IMC:

$$IMC = \frac{peso}{altura^2}$$

- retorne o valor calculado

Retorno e Tipo `bool`

Funções de verificação costumam retornar `bool`:

- `true` para condição satisfeita
- `false` caso contrário

Exemplo 4

Crie uma função que:

- receba um inteiro positivo `n`
- verifique se `n` é primo
- retorne `true` ou `false`

Exemplo 5

Um número `p` é primo de Sophie Germain quando:

- `p` é primo
- `2p + 1` também é primo

Crie uma função para verificar essa propriedade.

Escopo Local

Variáveis declaradas dentro da função:

- existem apenas durante aquela execução
- não podem ser acessadas fora da função

Chamada de Função

Uma função pode ser chamada:

- dentro do `main`
- dentro de outra função

Também pode ser usada em expressões, desde que retorne valor compatível.

Parâmetro com Valor Padrão

Em C++, podemos declarar:

```
void nome(int x = 1)
```

Quando nenhum argumento é informado, o padrão é usado.

Exemplo 6

Crie um procedimento para tabuada:

- recebe `n` como argumento
- usa `1` como padrão se nenhum valor for informado

Fechamento

Nesta aula:

- vimos funções e procedimentos
- trabalhamos parâmetros, retorno, escopo e valor padrão
- aplicamos modularização em problemas maiores

Próximo passo:

- resolver problemas completos com funções