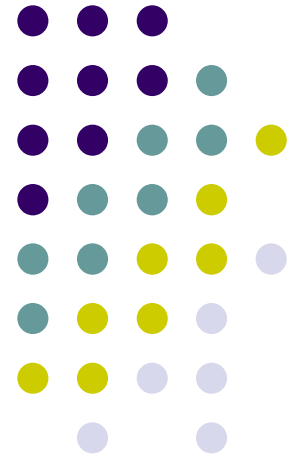


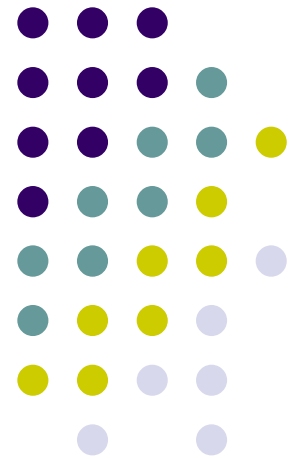
Fundamentos de Programação

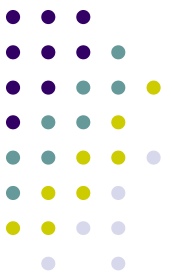
Instituto Federal de Mato Grosso

Tecnologia em Desenvolvimento
de Sistemas para Internet



Funções e Procedimentos





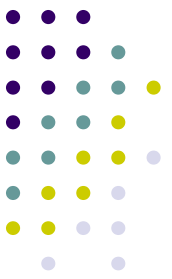
Funções e Procedimentos

- Os procedimentos e funções são nada mais do que subprogramas, isto é, pedaços de programas dentro de programas.
 - Mas, bem explorados, permitem a construção de códigos altamente elaborados. Existem três motivações para se usar subprogramas:
 - Modularidade;
 - reaproveitamento de código;
 - legibilidade do código.



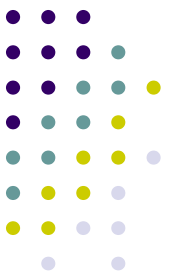
Modularidade

- A modularidade está relacionada com a capacidade de se escrever programas em pedaços de código que executam operações bem definidas.
 - Cada módulo possui variáveis e estruturas próprias, independentes do restante do programa.
 - A ideia é que modificações em trechos de código (necessárias para manutenção e continuidade de desenvolvimento) não causem reflexos no comportamento do resto do programa.



Reutilização de códigos

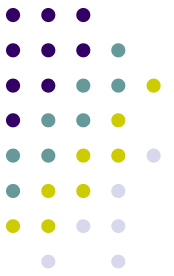
- Vez por outra nos deparamos com situações onde temos que escrever códigos muito, mas muito, parecidos em trechos diferentes do programa.
 - Ocorre frequentemente que o trecho é exatamente o mesmo.



Legibilidade e compreensão

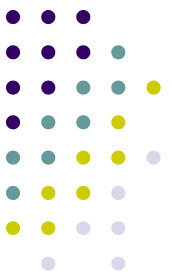
- Modularização e Reutilização de códigos, somados com o bom uso de nomes apropriados para os identificadores, indentação e uso racional de comentários no código, deve idealmente implicar em um código legível, isto é, compreensível para quem o lê e até mesmo para quem o escreveu

Noções fundamentais de funções e procedimentos



- Existem três perguntas que os estudantes fazem que precisam ser esclarecidas:
 - Quando usar **função** e quando usar **procedimento**?
 - Um procedimento difere de uma função basicamente por não ter um **valor de retorno**, já a função **retorna um valor**

```
1 programa
2 {
3     funcao inicio()
4     {
5         mensagem("Bem Vindo") // Chama o procedimento
6         escreva("O resultado do primeiro cálculo é: ", calcula (3.0, 4.0)) // Chama a função no escreva
7         escreva("\nO resultado do segundo cálculo é: ", calcula (7.0, 2.0), "\n") // Chama a função no escreva
8         mensagem("Tchau") // Chama o procedimento
9     }
10    funcao mensagem (cadeia texto) // procedimento não retorna valor algum
11    {
12        escreva ("\n", texto, "\n") // escreve a mensagem
13    }
14    funcao real calcula (real a, real b) // Função que realiza um cálculo e retorna o resultado
15    {
16        real resultado
17        resultado = a * a + b * b
18        retorne resultado
19    }
20 }
```

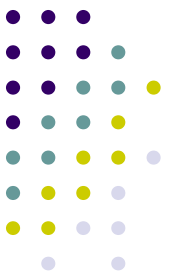
Exemplo de Procedimento

Identificador do
procedimento

Argumento ou
Parâmetro recebido

```
1 funcao mensagem (cadeia texto) // procedimento não retorna valor algum
2 {
3     escreva ("\n", texto, "\n") // escreve a mensagem
4 }
5
```

Bloco de comandos



Exemplo de Função

Tipo de Retorno da
informação

Identificador da
função

Argumentos ou
Parâmetros
recebidos

```
funcao real calcula (real a, real b) // Função que realiza um cálculo e retorna o resultado
15  {
16      real resultado
17      resultado = a * a + b * b
18      retorne resultado
19  }
```

Bloco de comandos