**Programação Aplicada**

**Modularização – Exercícios de Programação – Série 1**

1)Faça uma função que receba um valor real R, correspondente à medida do raio de um círculo, calcule e retorne a área do círculo (A = π R2). Elabore o algoritmo principal para testar sua função.

2)Elabore uma função que receba dois valores inteiros A e B e retorne verdadeiro se o valor B é um divisor de A ou retorne falso, caso contrário. Faça também o algoritmo principal para testar sua função.

3)Utilizando como base o código do exemplo 4, altere o código para que o algoritmo também faça o cálculo do número de arranjos de M elementos em grupos de P elementos. O número de arranjos é calculado pela fórmula

4)Faça um procedimento para identificar operações bancárias. O seu procedimento deve receber um valor inteiro correspondente ao código da operação e escrever na tela qual a descrição da operação de acordo com a tabela abaixo. Caso o código não esteja na tabela, deverá ser escrita a mensagem “Código não definido”. Faça também o algoritmo principal para testar a sua função.

|  |  |
| --- | --- |
| Código | Operação |
| 1 | Depósito em conta corrente |
| 2 | Depósito em conta poupança |
| 3 | Saque no caixa |
| 4 | Transferência Eletrônica Direta |
| 5 | Pagamento de conta |
| 6 | Pagamento de título |
| 7 | Pagamento de tributo |

5)Utilizando a função elaborada no exercício 2, faça uma nova função que receba um valor inteiro N e retorne o número de divisores de N existentes no intervalo de 1 até N. Por exemplo, se N for igual a 10, sua função deve retornar o número 4, pois 10 tem 4 divisores no intervalo de 1 até 10 (1, 2, 5 e 10). Faça o algoritmo principal para testar sua função.

6)Utilizando a função elaborada no exercício 5, crie uma nova função que receba um valor inteiro e retorne verdadeiro se esse valor for um número primo ou retorne falso, caso contrário. Faça o algoritmo principal para testar sua função.

7)Utilizando a função do algoritmo 6, crie um procedimento que leia um valor inteiro NUM escreva na tela todos os números primos no intervalo de 1 até NUM. Elabore o algoritmo principal para testar seu procedimento.