

## **Lista de Exercícios 1 – Subprogramas e parâmetros**

Sobre a lista:

Em algumas questões é solicitado que se faça um procedimento que *retorne* (e o itálico aqui é intencional) um ou mais valores. Mas procedimentos não retornam valores! Então o que se quis aqui foi que o procedimento tivesse parâmetros formais por referência para que os valores calculados dentro do procedimento pudessem voltar ao chamador. Isso pode acontecer ou porque o valor a ser retornado é composto (um vetor, um registro, por exemplo) ou que sejam vários valores, visto que o uso de uma função só permitiria retornar um único valor.

1. Faça o algoritmo de uma função que receba três valores numéricos reais e retorne aquele de maior valor.
2. Faça o algoritmo de um procedimento que receba três valores numéricos inteiros e escreva, na saída, todos os números compreendidos entre os valores dos dois primeiros parâmetros (inclusive), que sejam múltiplos do valor passado como terceiro parâmetro. Se o primeiro parâmetro for maior que o segundo a lista deve ser escrita em ordem crescente, e se for menor, em ordem decrescente.
3. Faça o algoritmo de uma função lógica que receba três valores inteiros e retorne VERDADEIRO se eles podem ser os valores dos lados de um triângulo, ou FALSO em caso contrário.
4. Faça o algoritmo de um procedimento que leia dois valores *string* que sejam garantidamente distintos, *retornando* estes dois valores, sendo que o primeiro parâmetro deve ser o menor e o segundo o maior.
5. Faça o algoritmo de um procedimento que receba um vetor de 100 números reais e *retorne* este vetor ordenado (usando qualquer dos métodos simples de ordenação). Defina, antes, um novo tipo de dado para este vetor.
6. Faça o algoritmo de um procedimento que receba um valor inteiro representando uma quantidade de segundos e *retorne* o equivalente em horas, minutos e segundos.
7. Faça o algoritmo de uma função inteira que receba uma *string* contendo um “número inteiro positivo” e retorne este inteiro. Caso o valor não possa ser convertido, deve ser retornado o valor  $-99999999$ .
8. Faça o algoritmo de uma função *string* que receba um número inteiro representando um número inteiro na base decimal e retorne a *string* com o equivalente valor escrito na base binária.
9. Faça o algoritmo de uma função lógica que receba uma *string* e retorne VERDADEIRO se ela é palíndrome, ou FALSO em caso contrário.
10. Faça o algoritmo de um função lógica que receba os três valores dos coeficientes de uma equação do segundo grau e *retorne* as suas duas raízes reais (independente se forem ou não iguais), retornando VERDADEIRO se existem raízes reais, ou FALSO em caso contrário.
11. Faça o algoritmo de um procedimento que seja uma versão alternativa e mais específica do comando `leia()`, que receba 3 parâmetros: uma variável inteira a ser lida, um valor mínimo e um valor máximo, e *retorne* um valor lido garantidamente entre o valor mínimo e máximo passados, ou seja, leia o valor até que ele esteja dentro dos limites permitidos.
12. Faça o algoritmo de uma função *string* que receba dois parâmetros: um caractere *c* e um número inteiro *n*, e retorne a *string* que tenha sequencialmente todos os *n* caracteres imprimíveis da tabela ASCII padrão a partir de *c*. Se houver menos que *n* caracteres nesta condição, não veja isto como problema e retorne a *string* com esse tamanho menor.