

# **Universidade Estadual de Goiás – UEG**

## **Sistemas de Informação**

### **Estrutura de Dados II**

#### **Revisão Recursividade**

##### **Exercício 1 – N-ésimo Número Triangular**

O número triangular de ordem “n” é dado pela soma:

$$T(n) = \frac{n(n+1)}{2}$$

Implemente recursivamente a soma:

$$T(n) = n + T(n-1), \quad T(1) = 1$$

##### **Exercício 2 – Potenciação ( $a^n$ )**

A potência  $a^N$  pode ser definida recursivamente como:

$$a^n = a \times a^{n-1}, \quad a^0 = 1$$

Implemente uma função recursiva que calcule a potência de um número inteiro.

##### **Exercício 3 – Contar os dígitos de um número**

Escreva uma função recursiva que receba um número inteiro e retorne quantos dígitos ele possui.

##### **Exercício 4. Verificar se uma palavra é palíndromo**

Escreva uma função recursiva que verifique se uma palavra é palíndromo (lida da mesma forma de trás para frente).

### **Exercício 5. Busca binária recursiva**

Implemente a busca binária de forma recursiva para encontrar um número dentro de um vetor ordenado.

### **Exercício 6. Contar vogais em uma string**

Elabore uma função recursiva que conte quantas vogais existem em uma string.

### **Exercício 7. Sequência de Collatz**

A sequência de Collatz é definida da seguinte forma:

- \* Se o número é par, divide-se por 2.
- \* Se é ímpar, multiplica-se por 3 e soma 1.

Repita até chegar a 1. Faça isso com recursividade.