

Recursividade

Deve ter em atenção que todos os exercícios desta ficha devem ser implementados no **projeto fornecido** no moodle.

1. Fatorial

O **fatorial** é uma função matemática que é calculada da seguinte forma:

$$fatorial(n) = \begin{cases} 1 & \Leftarrow n = 0 \\ n * fatorial(n - 1) & \Leftarrow n \in \mathbb{N} \end{cases}$$

- Implemente um algoritmo, recorrendo a recursividade, que produza o efeito da função fatorial;
- Execute a função para os valores: 5, 0, -5 e 21;
- Altere a implementação efetuada em a) de modo a utilizar a classe **EstatisticaDeChamadas**;
- Implemente uma função que apresente e devolva a estatística da execução (número de chamadas e tempo de execução) do fatorial dos valores indicados em b);
- Apresente uma janela de gráficos que demonstre a evolução do número de chamadas e do tempo de execução para o cálculo do fatorial com os valores de 1 a 100.

2. Fibonacci

A sequência de **Fibonacci** é representada pela seguinte fórmula:

$$fibonacci(n) = \begin{cases} n & \Leftarrow n \in \{0,1\} \\ fibonacci(n - 2) + fibonacci(n - 1) & \Leftarrow n \in \mathbb{N} \wedge n \geq 2 \end{cases}$$

- Implemente um algoritmo, recorrendo a recursividade, que produza o efeito da função fibonacci;
- Execute a função para os valores: 5, 0, -10, 40 e 45;

- c) Altere a implementação efetuada em a) de modo a utilizar a classe **EstatisticaDeChamadas**;
- d) Implemente uma função que apresente e devolva a estatística da execução (número de chamadas e tempo de execução) da função fibonacci para os valores indicados em b);
- e) Apresente uma janela de gráficos que demonstre a evolução do número de chamadas e do tempo de execução para o cálculo de fibonacci com os valores de 5 a 20 com incrementos de 3.

3. Torres de Hanói

O problema das **Torres de Hanói** (Figura 1) consiste em mover um conjunto de discos de uma base A para uma base C, com o auxílio de uma terceira base B, ordenados por tamanho (do maior para o menor, da base para o topo).

Para resolver o problema, devem-se movimentar os discos (um de cada vez) entre bases sem nunca colocar um disco maior sobre um disco menor.

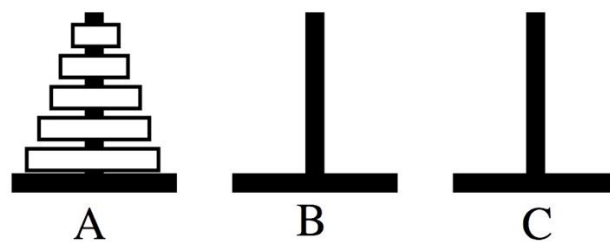


Figura 1. Torres de Hanói

- a) Implemente um algoritmo, recorrendo a recursividade, que resolva o problema das torres de Hanói;
- b) Execute a função para os valores: 3, 0, 5, 10, 15, 30, 31 e 32;
- c) Altere a implementação efetuada em a) de modo a utilizar a classe **EstatisticaDeChamadasEMovimentos**;
- d) Implemente uma função que apresente e devolva a estatística da execução (número de chamadas, número de movimentos e tempo de execução) de a) para os valores indicados em b);
- e) Apresente uma janela de gráficos que demonstre a evolução do número de chamadas, número de movimentos e do tempo de execução com os valores de 3 a 15.