

### GUIÃO 05 – ANÁLISE DA COMPLEXIDADE DE ALGORITMOS RECURSIVOS

Implemente os seguintes **algoritmos recursivos**, sem recorrer a funções de arredondamento (**floor** e **ceil**). Note que, considerando o quociente da divisão inteira, temos que  $n/2$  é igual a  $\lfloor \frac{n}{2} \rfloor$  e  $(n+1)/2$  é igual a  $\lceil \frac{n}{2} \rceil$ .

Determine o **número de chamadas recursivas** executadas por cada função.

- $T(n) = \begin{cases} 1, & \text{se } n = 1 \\ T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + n, & \text{se } n > 1 \end{cases}$
- $T(n) = \begin{cases} 1, & \text{se } n = 1 \\ T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + T\left(\left\lceil \frac{n}{2} \right\rceil\right) + n, & \text{se } n > 1 \end{cases}$
- $T(n) = \begin{cases} 1, & \text{se } n = 1 \\ T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + T\left(\left\lceil \frac{n}{2} \right\rceil\right) + n, & \text{se } n \text{ é ímpar} \\ 2 \times T\left(\frac{n}{2}\right) + n, & \text{se } n \text{ é par} \end{cases}$

Preencha a tabela, com o valor do resultado da função e o número de chamadas recursivas efetuadas, para os sucessivos valores de  $n$ , por exemplo, até 32 ou 64 ou 128.

N	1ª Função (N)	Nº de Chamadas	2ª Função (N)	Nº de Chamadas	3ª Função (N)	Nº de Chamadas
1	1	0	1	0	1	0
2						
3						
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						
15	26	3	37	5	49	7

Analisando os dados da tabela, qual é a **ordem de complexidade** de cada algoritmo?

Determine formalmente a **ordem de complexidade dos dois primeiros algoritmos**, obtendo **expressões matemáticas** exatas e simplificadas.

No caso do **terceiro algoritmo** indique para que valores de  $N$  se obtém o **melhor e o pior caso** e faça a respetiva análise da complexidade.