AULA 8 - ANÁLISE DA COMPLEXIDADE DE ALGORITMOS RECURSIVOS (NÚMERO DE SCHRÖDER)

- Implemente uma função recursiva para calcular o número de Schröder, definido pela seguinte relação de recorrência:

$$\text{Schröder}(n) = \begin{cases} 1, & \text{se } n = 0 \\ \text{Schröder}(n-1) + \sum_{i=0}^{n-1} \text{Schröder}(i) \times \text{Schröder}(n-i-1), & \text{se } n > 0 \end{cases}$$

Construa um programa para executar a função para sucessivos valores de n e que permita determinar experimentalmente a ordem de **complexidade das operações de multiplicação** do seu algoritmo. Efetue a análise empírica da complexidade construindo uma tabela com o número de multiplicações efetuadas para diferentes valores de n. Qual é a ordem de complexidade da função recursiva?

Uma forma de resolver problemas recursivos de maneira a evitar o cálculo repetido de valores, consiste em calcular os valores de baixo para cima, ou seja, de Schröder (0) para Schröder (n) e utilizar um *array* para manter os valores entretanto calculados. Este método designa-se por programação dinâmica e reduz o tempo de cálculo à custa da utilização de mais memória para armazenar valores intermédios.

- Usando a técnica de programação dinâmica, implemente uma função repetitiva alternativa e efetue a análise empírica da sua complexidade. Qual é a ordem de complexidade da função repetitiva?

- Faça a análise formal (no verso da folha) das implementações da função e confirme as ordens de complexidade obtidas experimentalmente.

N	Recursivo (N)	Nº de Multiplicações	Dinâmico (N)	Nº de Multiplicações
0	1	0	Igual	0
1	2	1	à versão	1
2	6	5	recursiva	3
3	22	20		6
4	90	76		10
5	394	285		15
6	1806	1065		21
7	8558	3976		28
8	41586	14840		36
9	206098	55385		45
10	1037718	206701		55
11	5293446	771420		66
12	27297738	2878980		78

O(N)	Exponencial	Quadrática

Na análise formal da solução recursiva – para simplificar a expressão recorrente – tenha em atenção que $\sum_{i=0}^{N-1} E(i) = \sum_{i=0}^{N-1} E(N-i-1)$.

NOME: Pedro Veloso Teixeira N° MEC: 84715

• Análise da versão recursiva

$$R_1(N) = \begin{cases} 0, & N = 0 \\ \underbrace{R_1(N-1)}_{\text{Schroder Rec }(N-1)} + \underbrace{N}_{\text{Ciclo For}} + \underbrace{\sum_{i=0}^{N-1} R_1(i) + \sum_{i=0}^{N-1} R_1(N-i-1)}_{\text{chamados recursivas dentro do ciclo for}} \end{cases}$$

$$\rightarrow R_1(N) = R_1(N-1) + N + \sum_{i=0}^{N-1} R_1(i) + \sum_{i=0}^{N-1} R_1(N-i-1) =$$

$$= R_1(N-1) + N + 2\sum_{i=0}^{N-1} R_1(i) \quad \text{por definição dada}$$

→ Então

$$R_1(N-1) = R_1(N-2) + (N-1) + 2\sum_{i=0}^{N-2} R_1(i)$$

→ Pelo que

$$R_1(N) - R_1(N-1) = R_1(N-1) - R_1(N-2) + \cancel{N} - \cancel{(N-1)} + 2\sum_{i=0}^{N-1} R_1(i) - 2\sum_{i=0}^{N-2} R_1(i) \Leftrightarrow$$

$$\Leftrightarrow R_1(N) = 2R_1(N-1) - R_1(N-2) + 1 + 2(R_1(N-1)) \Leftrightarrow$$

$$\Leftrightarrow R_1(N) = 4R_1(N-1) - R_1(N-2) + 1$$

$\rightarrow R_1(N) \leq 4R_1(N-1) - R_1(N-2)$. Seja $R_1(N) = 4R_1(N-1) - R_1(N-2)$

Equação característica: $x^2 - 4x + 1 = 0 \Leftrightarrow$

$$\Leftrightarrow x = \frac{4 \pm \sqrt{16-4}}{2} \Leftrightarrow x = \frac{4 \pm \sqrt{12}}{2} \quad \begin{cases} x \approx 3.73 \\ x \approx 0.27 \end{cases}$$

Pelo que $R_1(N) = C_1\, 3.73^n + C_2\, 0.27^n \rightarrow O(3.73^n)$ Exponencial

$(C_1, C_2 \in \mathbb{R})$

• Análise da versão de programação dinâmica

$$R_2(N) = \sum_{i=1}^{N} \sum_{j=0}^{i-1} 1 = \sum_{i=1}^{N} (i - \cancel{1} + \cancel{1}) = \sum_{i=1}^{N} i = \frac{N}{2}(1+N) = \frac{N^2+N}{2}$$

$O(n^2)$

Quadrático