Aula 6 - Análise da Complexidade de Algoritmos Recursivos

**\*\*\* Entregue, num ficheiro ZIP, este guião preenchido e o código desenvolvido \*\*\***

Implemente os seguintes **algoritmos recursivos** – **sem recorrer a funções de arredondamento** (floor e ceil) – e analise o **número de chamadas recursivas** executadas por cada algoritmo.

Deve utilizar **aritmética inteira**: n/4 é igual a e (n+3)/4 é igual a .

* **Preencha a tabela da página seguinte** com o resultado de cada função e o número de chamadas recursivas para os sucessivos valores de n.
* Analisando os dados da tabela, estabeleça uma ordem de complexidade para cada algoritmo.

|  |
| --- |
| **Denominado C1(n) , C2(n) e C3(n) como número de chamadas recursivas, tem-se**  **C1(n) tem a ordem de complexidade , logarítmica, fazendo a divisão C(4n)/C(n) obtém-se aproximadamente 1 que resulta numa ordem de complexidade do tipo**  **C2(n) tem a ordem de complexidade , inferior a linear , fazendo divisão C(4n)/C(n) obtém-se aproximadamente 2 que é igual a raiz quadrada de 4**  **C3(n) tema ordem de complexidade , inferior a linear visto que o nº de chamadas de T3(n) nunca é superior do nº de chamadas de T2(n) e ambas produzem o mesmo resultado**  ***Nota:***  **a analise da complexidade pode ser verificada no ficheiro Excel em anexo do zip file.** |

|  |
| --- |
| **Denominando C1(n) como número de chamadas recursivas para T1(n), tem-se:**  **usando agora o desenvolvimento telescópico**  **C1(n) = C1(floor(n/4))+1**  **C1(n) = C1(floor(n/42))+1+1**  **C1(n) = C1(floor(n/43))+1+1+1**  **C1(n) = C1(floor(n/4k))+k, em que n/4k = 1 , assim**  **C1(n) =** |

Escreva uma **expressão recorrente** para o **número de chamadas recursivas** efetuadas pela função **.** Obtenha, depois, uma **expressão exata e simplificada;** determine a sua **ordem de complexidade**. Compare a expressão obtida com os dados da **tabela**. Sugestão: use o **desenvolvimento telescópico**

**­­**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **n** | **T1(n)** | **Nº de Chamadas Recursivas** | **T2(n)** | **Nº de Chamadas Recursivas** | **T3(n)** | **Nº de Chamadas Recursivas** |
| 0 | **0** | **1** | **0** | **1** | **0** | **1** |
| 1 | **1** | **2** | **1** | **1** | **1** | **1** |
| 2 | **2** | **2** | **2** | **1** | **2** | **1** |
| 3 | **3** | **2** | **3** | **1** | **3** | **1** |
| 4 | **5** | **3** | **6** | **3** | **6** | **2** |
| 5 | **6** | **3** | **8** | **3** | **8** | **3** |
| 6 | **7** | **3** | **9** | **3** | **9** | **3** |
| 7 | **8** | **3** | **10** | **3** | **10** | **3** |
| 8 | **10** | **3** | **12** | **3** | **12** | **2** |
| 9 | **11** | **3** | **14** | **3** | **14** | **3** |
| 10 | **12** | **3** | **15** | **3** | **15** | **3** |
| 11 | **13** | **3** | **16** | **3** | **16** | **3** |
| 12 | **15** | **3** | **18** | **3** | **18** | **2** |
| 13 | **16** | **3** | **22** | **5** | **22** | **4** |
| 14 | **17** | **3** | **23** | **5** | **23** | **4** |
| 15 | **18** | **3** | **24** | **5** | **24** | **4** |
| 16 | **21** | **4** | **28** | **7** | **28** | **3** |
| 17 | **22** | **4** | **31** | **7** | **31** | **6** |
| 18 | **23** | **4** | **32** | **7** | **32** | **6** |
| 19 | **24** | **4** | **33** | **7** | **33** | **6** |
| 20 | **26** | **4** | **36** | **7** | **36** | **4** |
| 21 | **27** | **4** | **38** | **7** | **38** | **7** |
| 22 | **28** | **4** | **39** | **7** | **39** | **7** |
| 23 | **29** | **4** | **40** | **7** | **40** | **7** |
| 24 | **31** | **4** | **42** | **7** | **42** | **4** |
| 25 | **32** | **4** | **44** | **7** | **44** | **7** |
| 26 | **33** | **4** | **45** | **7** | **45** | **7** |
| 27 | **34** | **4** | **46** | **7** | **46** | **7** |
| 28 | **36** | **4** | **48** | **7** | **48** | **4** |

* Escreva uma **expressão recorrente** para o **número de chamadas recursivas** efetuadas pela função **. Considere o caso particular e** obtenha uma **expressão exata e simplificada;** determine a **ordem de complexidade** para esse caso particular. Compare a expressão obtida com os dados da **tabela**. Sugestão: use o **desenvolvimento telescópico** e confirme o resultado obtido usando o **Teorema Mestre**.

|  |
| --- |
| **Denominando C2(n) como número de chamadas recursivas para T2(n), tem-se:**  **C2 usando agora o desenvolvimento telescópico**  **C2(n) = 2C2(n/4)+2**  **C2(n) = 22C2(n/42)+4+2**  **C2(n) = 23C2(n/43)+23+22+2**  **C2(n) = 2kC2(n/4k)+2k+1-2, em que n/4k = 1 , assim**  **C2(n) = 3x2k-2=**  **Usando o teorema mestre, podemos comprovar o resultado obtido telescopicamente, tal que:**  **C2(n) = 2C2(n/4)+2 , em que a = 2, b = 4 e f(n), d = 0**    ***Resultou exatamente como queríamos confirmar pelo teorema mestre*** |

* Pode **generalizar a ordem de complexidade** que acabou de obter para todo o n? **Justifique.**

|  |
| --- |
| **Usando a regra da suavidade , a ordem de complexidade que obtive para casos particulares é válida para todo n** |

* Obtenha uma **expressão recorrente** para o **número de chamadas recursivas** efetuadas pela função

|  |
| --- |
| **Denominando C3(n) como número de chamadas recursivas para T3(n), tem-se:**  **C3**  ***Esta é a expressão recorrente encontrada do número de chamadas recursivas*** |

* **Considere o caso particular e** obtenha uma **expressão exata e simplificada;** determine a **ordem de complexidade** para esse caso particular. Compare a expressão obtida com os dados da **tabela**. Sugestão: use o **desenvolvimento telescópico** e confirme o resultado obtido usando o **Teorema Mestre**.

|  |
| --- |
| **usando agora o desenvolvimento telescópico , tem-se:**  **C3(n) = C2(n/4)+1**  **C3(n) = C2(n/42)+1+1**  **C3(n) = C2(n/43)+1+1+1**  **C3(n) = C3(n/4k)+k, em que n/4k = 1 , assim**  **C3(n) = =**  **Usando o teorema mestre, podemos comprovar o resultado obtido telescopicamente, tal que:**  **C3(n) = C3(n/4)+1 , em que a = 1, b = 4 e f(n), d = 0**    ***Resultou exatamente como queríamos confirmar pelo teorema mestre*** |

* Pode **generalizar a ordem de complexidade** que acabou de obter para todo o n? **Justifique.**

|  |
| --- |
| **Não se pode generalizar porque há casos particulares, por exemplo para números múltiplos de 4, o nº de chamadas recursivas é inferior à outros casos em que não sejam múltiplos de 4 e faz mais chamadas recursivas, por exemplo : para n = 13, 17 e 23, fazem 4, 6 e 7 chamadas recursivas respetivamente .** |

* Atendendo às **semelhanças entre e**  estabeleça uma **ordem de complexidade para . Justifique.**

|  |
| --- |
| **A ordem de complexidade de T3(n) é , a mesma de T2(n), porque calculam ambas o mesmo resultado , T3(n) faz menos operações recursivas em alguns casos, mas nunca mais chamadas recursivas do que T2(n).** |