Universidade Estadual Vale do Acaraú - UVA
Curso: Ciências da Computação
Disciplina: Construção e Análise de Algoritmos
Professor: Cláudio Carvalho

Lista 01 - Complexidade de Algoritmos

1. Suponha que um dado problema é resolvido, por um dado algoritmo, em $f(n)\ \mu s$, onde n indica o tamanho da entrada. Para cada função $f(n)$ a seguir, determine o maior tamanho de entrada desse problema que pode ser resolvida em $1s$.

   a) $f(n) = n$ b) $f(n) = n^2$ c) $f(n) = 2^n$ d) $f(n) = log_2 n$

2. Resolva as seguintes equações de recorrência:

   a) $T(n) = 4T(n/2) + n^2$ c) $T(n) = 2T(n/2) + n^2$ e) $T(n) = 2T(n-1) + 1$
   b) $T(n) = 4T(n/2) + n$ d) $T(n) = T(\sqrt{n}) + 1$ f) $T(n) = T(n/3) + 1$

3. Em relação a notação assintótica verifique se:
   (nos itens d e f, b é uma constante positiva diferente de 1.)

   a) $n^2 + 10n = O(n^2)$ c) $3^n = \Omega(2^n)$ e) $2^{n+1} = \Theta(2^n)$
   b) $n^2 = O(777n + 1345)$ d) $\log_b n = O(n)$. f) $\log_b n = \Theta(\log_2 n)$.

4. Sejam $f(n)$ e $g(n)$ funções assintoticamente positivas. Prove ou conteste as conjecturas:

   a) $f(n) = O(g(n))$ implica $g(n) = O(f(n))$
   b) $f(n) = O(g(n))$ é equivalente a $g(n) = \Omega(f(n))$
   c) $f(n) + g(n) = \Theta(\min(f(n), g(n)))$
   d) $f(n) = \Theta(f(n/2))$

5. Sabe-se que um computador C_1, executando um algoritmo A_1 de complexidade $\log_4^n$, resolve um problema de tamanho x em um tempo t milissegundos. Responda:

   a) Em quanto tempo C_1 resolve um problema de tamanho $16x$ usando A_1?
   b) Qual o tamanho da maior entrada que um computador C_2, 3 vezes mais rápido que C_1, resolve no mesmo tempo t com o algoritmo A_1?
   c) Em quanto tempo o computador C_1 resolve o mesmo problema de tamanho x, utilizando um algoritmo A_2 de complexidade $\log_2^n$?

6. Considerando apenas as operações de atribuição e comparação, apresente as funções de complexidade do melhor e do pior casos, para o *SelectionSort* e o *InsertionSort*.

7. Considerando $A(n)$ a função que determina o tempo de execução do algoritmo a seguir para uma entrada de tamanho n, responda:

   a) Calcule $A(n)$ em notação assintótica.

   b) Troque 8 por 7 no algoritmo e calcule novamente $A(n)$ em notação assintótica.

   c) Troque 8 por 16 e calcule novamente $A(n)$ em notação assintótica.

   Algoritmo Asteriscos(n)

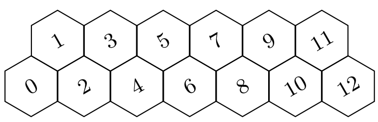
   se $n > 1$ **então**
       para $i \leftarrow 1$ **até** 8 **faça**
           Asteriscos($\lfloor n/2 \rfloor$)
           para $j \leftarrow 1$ **até** n^3 **faça**
               Imprima "$*$"

8. Dadas duas cadeias de caracteres S_1 e S_2. Diz-se que S_2 é um dos anagramas de S_1, e vice-versa, se elas possuem os mesmos caracteres. Por exemplo: 123, 321 e 132 são alguns dos anagramas de 123. Proponha um algoritmo de complexidade $\Theta(n)$ que receba duas cadeias de n dígitos e diga se elas são anagramas.

9. (Maratona de Programação da SBC - 2017) Todo inteiro positivo pode ser fatorado em números primos. Exemplo, $252 = 2^2 \times 3^2 \times 7$. Um inteiro é *despojado* se pode ser escrito como produto de dois ou mais primos distintos. Exemplo: $6 = 2 \times 3$ e $14 = 2 \times 7$ são despojados, mas $28 = 2^2 \times 7$, 1 e 17 não são despojados. Proponha um algoritmo que, dado um inteiro n ($1 \leq n \leq 10^{12}$), diga o número de divisores de n que são despojados. Mostre a complexidade de tempo do seu algoritmo.

10. Proponha um algoritmo que, dado um número inteiro n ($1 \leq n \leq 10^6$), apresente os números primos no intervalo de 1 a n. Apresente a complexidade do seu algoritmo.

11. (URI - Problema 1393 - adaptada) O caminho para a escola de Maria é pavimentado com lajotas hexagonais. A imagem abaixo mostra um caminho com peças de 0 a 12.

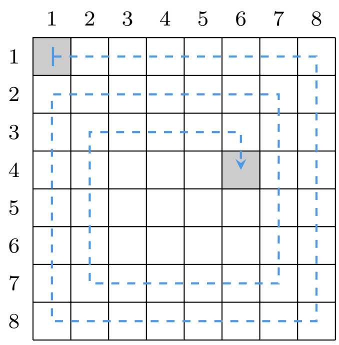
    

    Ao ir à escola, ela adota as seguintes regras: começa na lajota de número 0 e sempre passa para uma lajota vizinha cujo valor seja superior ao da que ela está pisando, até atingir a de maior valor. Maria não quer repetir sequências de passos nas lajotas. Por exemplo, há 5 sequênias distinas para lajotas numeradas de 0 a 4: $\langle 0, 1, 2, 3, 4 \rangle$, $\langle 0, 1, 2, 4 \rangle$, $\langle 0, 1, 3, 4 \rangle$, $\langle 0, 2, 3, 4 \rangle$ e $\langle 0, 2, 4 \rangle$. Dado um caminho com lajotas hexagonais numeradas de 0 a n, $1 \leq n \leq 40$, proponha um algoritmo que diga em tempo $O(n)$ o número de maneiras distintas que levam da lajota 0 à n.

12. (URI - Problema 2867) Dados dois inteiros n e m, com $1 \leq m, n \leq 100$, elabore um algoritmo que calcule a quantidade de dígitos do número n^m.

13. Considere duas filas, uma de armários $A_1, \ldots, A_n$ (inicialmente fechados), e outra de pessoas $P_1, \ldots, P_n$. Cada pessoa P_i, $1 \leq i \leq n$, deve passar pelos armários e mudar o status (aberto $\leftrightarrow$ fechado) de cada armário A_j, $1 \leq j \leq n$, sempre que j for um múltiplo de i. Proponha um algoritmo que leia uma quantidade n de armários e de pessoas, $1 \leq n \leq 10^8$, e diga quantos armários ficarão abertos após a passagem da última pessoa. Analise a complexidade do seu algoritmo, em notação assintótica.

14. (URI - Problema 1935) Dado um tabuleiro de dimensões $n \times n$, deseja-se colocar grãos de feijão, um em cada quadrado, seguindo uma espiral, começando na posição $(1, 1)$, indo para a direita, para baixo, para a esquerda e para cima. O sentido do movimento só muda quando atingir a borda do tabuleiro, ou quando a próxima posição estiver ocupada. Esse padrão deve se repetir até que b grãos de feijão sejam colocados. Dados os valores n $(1 \leq n \leq 2^{30})$ e b $(1 \leq b \leq n^2)$, proponha um algoritmo que diga, em tempo $O(1)$, a linha e a coluna em que ficará o último grão. Exemplo: para $n = 8$ e $b = 53$, conforme a figura a seguir, o último grão ficará na posição $(4, 6)$.



15. (URI - Problema 3087 - adaptada) Considere uma matriz A de ordem $n \times n$, com n ímpar, que deve ser preenchida com a sequência de valores de 1 a n^2, a partir do centro, em espiral. Proponha um algoritmo que, dados a ordem n da matriz A e um número s, $n \in \{1, 3, \ldots, 99\}$ e $1 \leq s \leq n^2$, calcule a posição (linha, coluna) de s em A. No exemplo a seguir, para $n = 5$ e $s = 6$, s está na posição $(2, 1)$.

	0	1	2	3	4
0	21	22	23	24	25
1	20	7	8	9	10
2	19	6	1	2	11
3	18	5	4	3	12
4	17	16	15	14	13

16. (URI - Problema 1801 - adaptada) Dado um número inteiro n, com $1 \leq n \leq 2^{12} - 1$, proponha um algoritmo que determine o número de permutações p de n tais que p não possui zeros à esquerda e o número $p + n$ é um quadrado perfeito. Faça a análise de complexidade do seu algoritmo. Por exemplo, para $n = 152784$, há 8 possíveis valores para p. São eles: 182457, 418752, 527841, 578241, 815472, 845217, 857241 e 875412.