

Universidade Federal do Pará
Instituto de Ciências Exatas e Naturais
Faculdade de Computação
Análise de Algoritmos

Exercícios
Conceitos Básicos e Notações Assintóticas

1. Considere duas funções $f(n) = 0,5n^2 - 3n$ e $g(n) = n^2$. Agora, usando a notação Big-O, mostre que $f(n)$ é $O(g(n))$.

2. Dado dois algoritmos A e B que resolvem o mesmo problema e possuem complexidade no tempo $8n^2$ e n^3 , respectivamente, responda os itens abaixo:

a. Qual é o maior valor de entrada n para o qual o algoritmo B é mais eficiente que o algoritmo A ?

b. Na sua opinião, qual algoritmo é mais eficiente?

3. Determine a complexidade no tempo do algoritmo apresentado abaixo que promete encontrar os elementos mínimo e máximo de um vetor de entrada V de tamanho n .

MaxMin (V, n)

1. max = V[1]

2. min = V[1]

3. para i = 2 até n faça

4. se V[i] > max então max = V[i]

5. se V[i] < min então min = V[i]

Agora, responda os itens a seguir:

a. O algoritmo *MaxMin* para e é correto?

b. Existe melhor e pior caso?

c. O algoritmo *MaxMin* é eficiente?

4. Usando notação assintótica, descreva a complexidade no tempo do algoritmo abaixo. Ele recebe dois vetores A e V de tamanhos n e w , respectivamente.

```
TESTE (A, V, n, w)
1. para i = 1 até n faça
2.   para j = 1 até w faça
3.     V[j] = A[i] + 2
4. max = V[1]
5. para j = 2 até w faça
6.   se max < V[j] então max = V[j]
```

5. Mostre que $7x^2$ é $O(x^3)$. Também é verdade que x^3 é $O(7x^2)$? Essas funções são assintoticamente equivalentes?

6. Mostre que x^2 é $\Omega(x)$. E a recíproca é verdadeira, ou seja, também é verdade que x é $\Omega(x^2)$? Essas funções são assintoticamente equivalentes?

7. As afirmações abaixo são verdadeiras ou falsas? Justifique.

a. $2n^2 + 1000$ é $\Omega(n^2)$.

b. $\log(n^2)$ é $\omega(\log(n))$.

c. 2^{n+1} é $O(2^n)$.

d. 2^{2n} é $O(2^n)$.

8. Um algoritmo tradicional possui complexidade $n^{1,5}$, enquanto um novo algoritmo proposto é da ordem de $n \log n$:

$$f(n) = n^{1,5}$$

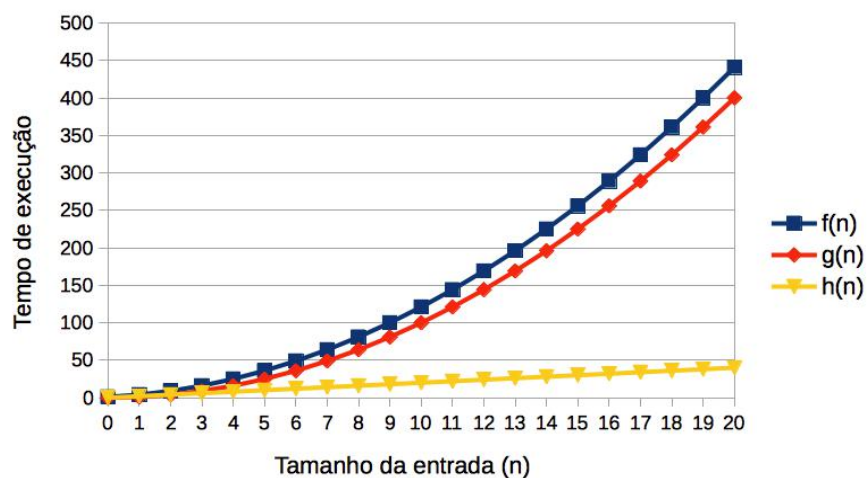
$$g(n) = n \log n$$

Qual algoritmo adotar? Por quê?

9. Considerando $f \equiv f(n)$, $g \equiv g(n)$ e k uma constante, determine se as sentenças abaixo são verdadeiras ou falsas. Caso sejam falsas reescreva corretamente.

- $O(f + g) = O(f) + O(g)$.
- $O(f.g) = O(f).O(g)$.
- $O(k.g) = k.O(g) = O(g)$.
- Se $f = O(g)$, então $g = \Omega(f)$.
- Se $f = O(g)$ e $g = O(h)$, então $f = O(h)$.

10. Observe as funções $f(n) = n^2 + 2n$, $g(n) = n^2$ e $h(n) = 2n + 1$ representadas no gráfico abaixo.



Assinale a afirmativa correta sobre o crescimento assintótico dessas funções.

- $f(n) = O(h(n))$ e $f(n) = \omega(g(n))$.
- $g(n) = \Omega(f(n))$ e $f(n) = \Theta(h(n))$.
- $h(n) = \omega(g(n))$ e $g(n) = \Theta(f(n))$.
- $g(n) = O(f(n))$ e $g(n) = \Omega(h(n))$.
- $h(n) = \Omega(f(n))$ e $h(n) = O(g(n))$.

11. Considerando que todos os logaritmos tem base 2, assinale a afirmativa abaixo que é **FALSA**.

- (a) $n^2 + \log(n) + 5 = o(n^3)$.
- (b) $2n\log(n) + 4n + 10\sqrt{n} = 2n\log(n) + O(n^2)$.
- (c) $n^2 + 2 = \Theta(4^{\log(n)})$.
- (d) $n^{\frac{1}{\log(n)}} = \Theta(n)$.
- (e) $\log(n) + \sqrt{n} = O(n)$.

12. [POSCOMP 2003] Qual é o número mínimo de comparações necessário para encontrar o menor elemento de um conjunto qualquer não ordenado de n elementos?

- (a) 1
- (b) $n - 1$
- (c) n
- (d) $n + 1$
- (e) $n \log n$

13. A função SORT abaixo ordena de forma crescente um vetor A de n elementos.

`SORT (A, n)`

- 1. `para j = 1 até n - 1 faça`
- 2. `menor = j`
- 3. `para i = j + 1 até n faça`
- 4. `se A[i] < A[menor] então menor = i`
- 5. `aux = A[menor]`
- 6. `A[menor] = A[j]`
- 7. `A[j] = aux`

Dado que $T(n)$ é o tempo de execução da função SORT para as entradas A e n , é possível afirmar que a ordem de $T(n)$ é

- (a) $T(n) = O(1)$.
- (b) $T(n) = O(\log(n))$.
- (c) $T(n) = O(n)$.
- (d) $T(n) = o(n^2)$.
- (e) $T(n) = O(n^2)$.

14. [POSCOMP 2010] Considere dois algoritmos $A1$ e $A2$, cujas funções de custo são, respectivamente, $T1(n) = n^2 - n + 1$ e $T2(n) = 6n \log_2 n + 2n$. Para simplificar a análise, assuma que $n > 0$ é sempre uma potência de 2. Com relação ao enunciado, assinale a alternativa correta.

- (a) $T1(n) = \Theta(n^2)$ e $T2(n) = \Theta(n \log n)$, então $A2$ é sempre mais eficiente que $A1$.
- (b) O limite superior $T1(n) = O(n^3)$ é correto e assintoticamente restrito.
- (c) O limite inferior $T2(n) = \Omega(n^3)$ é correto e assintoticamente restrito.
- (d) $T1$ e $T2$ são assintoticamente equivalentes.
- (e) $A1$ é mais eficiente que $A2$, para n suficientemente pequeno.

15. [POSCOMP 2004] Um algoritmo é executado em 10 segundos para uma entrada de tamanho 50. Se o algoritmo é quadrático, quanto tempo em segundos ele gastará, aproximadamente, no mesmo computador, se a entrada tiver tamanho 100?

- (a) 10
- (b) 20
- (c) 40
- (d) 100
- (e) 500

16. [POSCOMP 2022] Considere as funções a seguir:

$$f1(n) = O(n)$$

$$f2(n) = O(n!)$$

$$f3(n) = O(2^n)$$

$$f4(n) = O(n^2)$$

A ordem dessas funções, por ordem crescente de taxa de crescimento, é:

- (A) $f2 - f1 - f3 - f4$.
- (B) $f3 - f2 - f4 - f1$.
- (C) $f1 - f4 - f3 - f2$.
- (D) $f1 - f4 - f2 - f3$.
- (E) $f4 - f3 - f1 - f2$.

17. [POSCOMP 2003] Quais das seguintes igualdades são verdadeiras?

- I. $n^2 = O(n^3)$
- II. $2n + 1 = O(n^2)$
- III. $n^3 = O(n^2)$
- IV. $3n + 5n \log(n) = O(n)$
- V. $\log(n) + \sqrt{n} = O(n)$

- (a) Somente I e II.
- (b) Somente II, III e IV.
- (c) Somente III, IV e V.
- (d) Somente I, II e V.
- (e) Somente I, III e IV.

18. [POSCOMP 2002] Qual das seguintes afirmações sobre crescimento assintótico de funções **NÃO** é verdadeira?

- (a) $2n^2 + 3n + 1 = O(n^2)$
- (b) Se $f(n) = O(g(n))$, então $g(n) = O(f(n))$
- (c) $\log(n^2) = O(\log(n))$
- (d) Se $f(n) = O(g(n))$ e $g(n) = O(h(n))$, então $f(n) = O(h(n))$
- (e) $2^{n+1} = O(2^n)$

19. [POSCOMP 2015] Sejam $T_1(n) = 100n + 15$, $T_2(n) = 10n^2 + 2n$ e $T_3(n) = 0,5n^3 + n^2 + 3$ as equações que descrevem a complexidade de tempo dos algoritmos *Alg1*, *Alg2* e *Alg3*, respectivamente, para entradas de números inteiros de tamanho $n > 0$. Assinale a alternativa correta.

- (a) As complexidades assintóticas de *Alg1*, *Alg2* e *Alg3* em notação Big-O estão, respectivamente, em $O(n)$, $O(n^2)$, $O(n^3)$.
- (b) As complexidades assintóticas de *Alg1*, *Alg2* e *Alg3* em notação Big-O estão, respectivamente, em $O(n)$, $O(n^2)$, $O(n^2)$.
- (c) As complexidades assintóticas de *Alg1*, *Alg2* e *Alg3* em notação Big-O estão, respectivamente, em $O(100)$, $O(10)$, $O(0,5)$.
- (d) *Alg2* e *Alg3* pertencem às mesmas classes de complexidade assintótica.
- (e) *Alg1* e *Alg2* pertencem às mesmas classes de complexidade assintótica.

20. [POSCOMP 2011] Sejam $T_A(n)$ e $T_B(n)$ os tempos de execução de pior caso de dois algoritmos A e B propostos para um mesmo problema computacional, em função de um certo parâmetro n . Dizemos que o algoritmo A é mais eficiente que o algoritmo B assintoticamente no pior caso quando

- (a) $T_A(n) = o(T_B(n))$.
- (b) $T_B(n) = o(T_A(n))$.
- (c) $T_A(n) = O(T_B(n))$.
- (d) $T_B(n) = O(T_A(n))$.
- (e) $T_A(n) = \Theta(T_B(n))$.

21. [POSCOMP 2016] Um algoritmo tem complexidade $O(3m^3 + 2mn^2 + n^2 + 10m + m^2)$. Uma maneira simplificada de representar a complexidade desse algoritmo é

- (a) $O(m^3 + mn^2)$.
- (b) $O(m^3)$.
- (c) $O(m^2)$.
- (d) $O(mn^2)$.
- (e) $O(m^3 + n^2)$.

22. [POSCOMP 2018] Dado o trecho de código

```

1. int i, j, c;
2. c = 1;
3. for (i = 1; i < n; i = i*2) {
4.     for (j = 1; j <= n; j++) {
5.         c = c + 1;
6.     }
7. }
```

Assumindo que a instrução $c = c + 1$ é $O(1)$, a expressão que melhor define a ordem de complexidade desse trecho é

- (a) $O(n \log n)$
- (b) $O(\log n)$
- (c) $O(n)$
- (d) $O(n^2)$
- (e) $O(\sqrt{n})$

23. [POSCOMP 2018] Para medir o custo de execução de um algoritmo, é comum definir uma função de complexidade f , em que $f(n)$ é a medida de tempo necessário para executar um algoritmo para um problema de tamanho n . Considere as afirmações abaixo sobre funções de complexidade:

I. Se $f(n)$ é uma medida de quantidade de tempo necessário para executar um algoritmo em um problema de tamanho n , então f é chamada função de complexidade de tempo.

II. Se $f(n)$ é uma medida de quantidade de memória necessária para executar um algoritmo de tamanho n , então f é chamada função de complexidade de espaço.

III. A complexidade de tempo não representa o tempo diretamente, contudo, é estimada pelo número de vezes que determinada operação relevante é executada.

Quais estão corretas?

- (a) Apenas I.
- (b) Apenas II.
- (c) Apenas III.
- (d) Apenas I e II.
- (e) I, II e III.