

Lista 2 - MC458 - Gabarito

Tema: Análise Assintótica

1. Sejam f e g funções reais não decrescentes definidas nos inteiros positivos, com $f(n)$ e $g(n)$ maiores ou iguais a 1 para todo n . Assuma que $f(n) = O(g(n))$, e seja c uma constante positiva. A afirmação $f(n) \cdot \log_2(f(n)^c) = O(g(n) \cdot \log_2(g(n)))$ é verdadeira?

Resposta: d) Às vezes sim, às vezes não, dependendo das funções f e g .

Demonstração/Justificativa:

Usando propriedades de logaritmo, reescrevemos o termo da esquerda como $f(n) \cdot \log_2(f(n)^c) = c \cdot f(n) \log_2(f(n))$.

I. Caso em que a afirmação é FALSA (Contraexemplo):

Seja $g(n) = 1$ para todo n . Como $g(n) \geq 1$ e é constante, satisfaz todas as hipóteses. O lado direito resulta em:

$$g(n) \log_2(g(n)) = 1 \cdot \log_2(1) = 0$$

Pelo enunciado, $f(n) \geq 1$. Como $f(n) = O(g(n))$, existe $M > 0$ tal que $1 \leq f(n) \leq M \cdot 1$. Escolhendo a função constante $f(n) = 2$ (com $M = 2$), temos:

$$c \cdot f(n) \log_2(f(n)) = c \cdot 2 \cdot \log_2(2) = 2c > 0$$

Como $2c > 0$, não existe constante $C > 0$ tal que $2c \leq C \cdot 0$. Logo, para este caso, a relação O falha.

II. Caso em que a afirmação é VERDADEIRA (quando $g(n) \rightarrow \infty$):

Como $f(n) = O(g(n))$, existem $c_0 > 0$ e n_1 tais que $f(n) \leq c_0 \cdot g(n)$ para todo $n \geq n_1$. Aplicando o logaritmo em ambos os lados:

$$\log_2(f(n)) \leq \log_2(c_0 \cdot g(n)) = \log_2(c_0) + \log_2(g(n))$$

Como $g(n) \rightarrow \infty$, o termo $\log_2(g(n))$ cresce sem limites, enquanto $\log_2(c_0)$ é uma constante fixa.

- Se $c_0 \leq 1$, então $\log_2(c_0) \leq 0$. Assim, $\log_2(c_0) + \log_2(g(n)) \leq 1 \cdot \log_2(g(n))$ (basta tomar $k = 1$).
- Se $c_0 > 1$, como $g(n) \rightarrow \infty$, existe n_2 a partir do qual $g(n) \geq c_0$. Isso garante que $\log_2(g(n)) \geq \log_2(c_0)$. Substituindo o termo constante:

$$\log_2(c_0) + \log_2(g(n)) \leq \log_2(g(n)) + \log_2(g(n)) = 2 \cdot \log_2(g(n))$$

Portanto, escolhendo $k = 2$ e $n_0 = \max(n_1, n_2)$, temos $\log_2(f(n)) \leq k \cdot \log_2(g(n))$.

Multiplicação dos termos para o resultado final:

Multiplicando as duas desigualdades obtidas ($f(n) \leq c_0 g(n)$ e $\log_2(f(n)) \leq k \log_2(g(n))$):

$$f(n) \cdot \log_2(f(n)) \leq (c_0 \cdot g(n)) \cdot (k \cdot \log_2(g(n))) = (c_0 \cdot k) \cdot [g(n) \log_2(g(n))]$$

Multiplicando pela constante $c > 0$:

$$f(n) \cdot \log_2(f(n)^c) = c \cdot f(n) \log_2(f(n)) \leq (c \cdot c_0 \cdot k) \cdot [g(n) \log_2(g(n))]$$

Tomando a constante $C = c \cdot c_0 \cdot k$, provamos que $f(n) \cdot \log_2(f(n)^c) = O(g(n) \cdot \log_2(g(n)))$.

Como a veracidade depende de $g(n)$ tender ao infinito ou ser constante, a afirmação vale às vezes sim, às vezes não, dependendo das funções f e g .

2. Assuma novamente duas funções positivas e não decrescentes f e g tais que $f(n) = O(g(n))$. É verdade que $2^{f(n)} = O(2^{g(n)})$?

Resposta:

c) Às vezes sim, às vezes não, dependendo das funções f e g .

d) Sim, sempre que $f(n) \leq g(n)$ para todo n suficientemente grande.

Demonstração/Justificativa:

Fundamentação Teórica (Critério do Limite para a Notação O):

Sejam $a(n)$ e $b(n)$ duas funções estritamente positivas. O *Teste do Limite* estabelece que:

$$L = \lim_{n \rightarrow \infty} \frac{a(n)}{b(n)}$$

- i) Se $L < \infty$ (incluindo $L = 0$), então $a(n) = O(b(n))$, pois a razão permanece limitada superiormente por uma constante à medida que n cresce.
- ii) Se $L = \infty$, então $a(n) \notin O(b(n))$, pois $a(n)$ cresce assintoticamente mais rápido do que qualquer múltiplo constante de $b(n)$.

Análise das Alternativas:

I. Justificativa da Alternativa c) (A relação às vezes vale e às vezes não):

- **Caso em que a afirmação é FALSO (Não vale):**

Sejam $f(n) = 2n$ e $g(n) = n$. Temos $f(n) = O(g(n))$ (pois $2n \leq 2 \cdot n$). Aplicando o Teste do Limite para as exponenciais:

$$\lim_{n \rightarrow \infty} \frac{2^{f(n)}}{2^{g(n)}} = \lim_{n \rightarrow \infty} \frac{2^{2n}}{2^n} = \lim_{n \rightarrow \infty} 2^n = \infty$$

Pelo item (ii) do Teorema do Limite, $2^{2n} \notin O(2^n)$.

- **Caso em que a afirmação é VERDADEIRA (Vale):**

Sejam $f(n) = n + 1$ e $g(n) = n$. Temos $f(n) = O(g(n))$ (pois $n + 1 \leq 2n$ para $n \geq 1$). Aplicando o Teste do Limite para as exponenciais:

$$\lim_{n \rightarrow \infty} \frac{2^{f(n)}}{2^{g(n)}} = \lim_{n \rightarrow \infty} \frac{2^{n+1}}{2^n} = \lim_{n \rightarrow \infty} 2 = 2 < \infty$$

Pelo item (i) do Teorema do Limite, como o limite é uma constante finita ($L = 2$), conclui-se que $2^{n+1} = O(2^n)$.

Como existem pares de funções que satisfazem a propriedade e outros que a violam, a afirmação depende das funções escolhidas.

II. Justificativa da Alternativa d) (Condição suficiente):

Se $f(n) \leq g(n)$ para todo $n \geq n_0$, aplicando a propriedade da monotonicidade da função exponencial 2^x (que é estritamente crescente), temos:

$$2^{f(n)} \leq 2^{g(n)} = 1 \cdot 2^{g(n)}, \quad \forall n \geq n_0$$

Pela definição formal de O ($a(n) \leq c \cdot b(n)$), basta escolher a constante $c = 1$ e o mesmo n_0 . Logo, a relação $2^{f(n)} = O(2^{g(n)})$ é garantida sempre que $f(n) \leq g(n)$ para n grande.

3. Organize as seguintes funções em ordem crescente de taxa de crescimento, de modo que $g(n)$ venha logo após $f(n)$ na sua lista se, e somente se, $f(n) = O(g(n))$:

Resposta:

$$2\sqrt{\log_2 n} \prec \sqrt{n} \prec n^{1.5} \prec n^{5/3} \prec 10^n$$

Demonstração/Justificativa:

Análise detalhada de cada termo:

I. Reescrita de $2\sqrt{\log_2 n}$ via propriedade de logaritmo:

Pela propriedade fundamental dos logaritmos, sabemos que $2^{\log_2 n} = n$. Reescrevendo o expoente $\sqrt{\log_2 n}$ como $\frac{\log_2 n}{\sqrt{\log_2 n}}$, podemos reescrever a função da seguinte forma:

$$2\sqrt{\log_2 n} = 2^{\frac{\log_2 n}{\sqrt{\log_2 n}}} = (2^{\log_2 n})^{\frac{1}{\sqrt{\log_2 n}}} = n^{\frac{1}{\sqrt{\log_2 n}}}$$

Como $\lim_{n \rightarrow \infty} \frac{1}{\sqrt{\log_2 n}} = 0$, o expoente de n tende a zero quando $n \rightarrow \infty$. Logo, esta função cresce assintoticamente mais devagar do que qualquer polinômio n^ϵ com constante $\epsilon > 0$. Em particular:

$$\lim_{n \rightarrow \infty} \frac{2\sqrt{\log_2 n}}{\sqrt{n}} = \lim_{n \rightarrow \infty} \frac{n^{\frac{1}{\sqrt{\log_2 n}}}}{n^{0.5}} = \lim_{n \rightarrow \infty} n^{\frac{1}{\sqrt{\log_2 n}} - 0.5} = 0 \implies 2\sqrt{\log_2 n} = o(\sqrt{n})$$

II. Comparação das potências polinomiais ($\sqrt{n}$, $n^{1.5}$, $n^{5/3}$):

Para funções do tipo n^k , a taxa de crescimento é determinada estritamente pela magnitude do expoente k :

- $\sqrt{n} = n^{0.5}$ (expoente 0.5)
- $n^{1.5} = n^{3/2}$ (expoente 1.5)
- $n^{5/3} \approx n^{1.667}$ (expoente 1.667)

Como $0.5 < 1.5 < \frac{5}{3}$, temos que $\sqrt{n} = o(n^{1.5})$ e $n^{1.5} = o(n^{5/3})$.

III. Comparação com a função exponencial (10^n):

Funções exponenciais de base maior que 1 crescem estritamente mais rápido do que qualquer função polinomial n^k . Assim, para qualquer $k > 0$:

$$\lim_{n \rightarrow \infty} \frac{n^{5/3}}{10^n} = 0 \implies n^{5/3} = o(10^n)$$

Portanto, a ordenação crescente correta da taxa de crescimento é:

$$2\sqrt{\log_2 n} \prec \sqrt{n} \prec n^{1.5} \prec n^{5/3} \prec 10^n$$

4. Provas formais pela definição:

Fundamentação Teórica (Definições Formais de Notação Assintótica):

Sejam $f(n)$ e $g(n)$ funções assintoticamente não negativas definidas nos inteiros positivos.

- **Notação Big-O (O):** $f(n) = O(g(n))$ se e somente se existem constantes reais $c > 0$ e $n_0 \geq 0$ tais que:

$$0 \leq f(n) \leq c \cdot g(n), \quad \forall n \geq n_0$$

- **Notação Big-Omega (Ω):** $f(n) = \Omega(g(n))$ se e somente se existem constantes reais $c > 0$ e $n_0 \geq 0$ tais que:

$$0 \leq c \cdot g(n) \leq f(n), \quad \forall n \geq n_0$$

- **Notação Big-Theta (Θ):** $f(n) = \Theta(g(n))$ se e somente se existem constantes reais $c_1, c_2 > 0$ e $n_0 \geq 0$ tais que:

$$0 \leq c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n), \quad \forall n \geq n_0$$

Nota: $f(n) = \Theta(g(n))$ se e somente se $f(n) = O(g(n))$ e $f(n) = \Omega(g(n))$.

- a) **Prove que** $5n^2 - 3n + 20 = \Theta(n^2)$.

Demonstração/Justificativa:

Pela definição de Θ , precisamos encontrar constantes $c_1, c_2 > 0$ e $n_0 \geq 0$ tais que:

$$c_1 n^2 \leq 5n^2 - 3n + 20 \leq c_2 n^2, \quad \forall n \geq n_0$$

Demonstraremos os limites superior e inferior separadamente:

- **1. Limite Superior (c_2):**

Para todo $n \geq 1$, sabemos que $-3n \leq 0$ e que $20 \leq 20n^2$. Portanto:

$$5n^2 - 3n + 20 \leq 5n^2 + 0 + 20n^2 = 25n^2$$

Logo, podemos escolher $c_2 = 25$ para $n \geq 1$.

- **2. Limite Inferior (c_1):**

Como $20 > 0$, temos a desigualdade $5n^2 - 3n + 20 \geq 5n^2 - 3n$.

Para todo $n \geq 1$, vale que $n \leq n^2$, o que implica $-3n \geq -3n^2$. Substituindo:

$$5n^2 - 3n + 20 \geq 5n^2 - 3n^2 = 2n^2$$

Logo, podemos escolher $c_1 = 2$ para $n \geq 1$.

Conclusão: Como $2n^2 \leq 5n^2 - 3n + 20 \leq 25n^2$ vale para todo $n \geq 1$, a afirmação é verdadeira com $c_1 = 2$, $c_2 = 25$ e $n_0 = 1$.

- b) **Prove por contradição que** $n^3 \notin O(n^2)$.

Demonstração/Justificativa:

Suponha por absurdo que $n^3 = O(n^2)$. Pela definição de O , existiriam constantes $c > 0$ e $n_0 \geq 0$ tais que:

$$n^3 \leq c \cdot n^2, \quad \forall n \geq n_0$$

Dividindo ambos os lados por n^2 (para $n > 0$):

$$n \leq c, \quad \forall n \geq n_0$$

Isso exigiria que a variável n fosse limitada superiormente pela constante c para todo $n \geq n_0$, o que é uma contradição com o fato de n crescer arbitrariamente (basta tomar $n = \max(n_0, \lfloor c \rfloor + 1)$ para ter $n > c$). Logo, $n^3 \notin O(n^2)$.

c) **Prove que** $100n \log_2 n = O(n^2)$.

Demonstração/Justificativa:

Para todo $n \geq 1$, vale a propriedade de que $\log_2 n \leq n$.

Multiplicando ambos os lados por $100n$ (termo estritamente positivo para $n \geq 1$):

$$100n \log_2 n \leq 100n \cdot n = 100n^2, \quad \forall n \geq 1$$

Pela definição de O , a relação $100n \log_2 n \leq c \cdot n^2$ está satisfeita com $c = 100$ e $n_0 = 1$.

d) **Prove que** $\frac{1}{2}n^2 - 3n = \Omega(n^2)$.

Demonstração/Justificativa:

Pela definição de Ω , precisamos encontrar constantes $c > 0$ e $n_0 \geq 0$ tais que:

$$\frac{1}{2}n^2 - 3n \geq c \cdot n^2, \quad \forall n \geq n_0$$

Desenvolvimento e dedução das constantes:

1. Dividindo a desigualdade por n^2 (para $n > 0$), queremos encontrar $c > 0$ tal que:

$$\frac{1}{2} - \frac{3}{n} \geq c$$

2. Para garantir um valor positivo para c , devemos restringir n de forma que o termo negativo $\frac{3}{n}$ seja menor do que $\frac{1}{2}$.

3. Vamos escolher, por exemplo, limitar $\frac{3}{n}$ para ser no máximo a metade de $\frac{1}{2}$, ou seja, $\frac{3}{n} \leq \frac{1}{4}$.

4. Isolando n na desigualdade $\frac{3}{n} \leq \frac{1}{4}$:

$$3 \leq \frac{1}{4}n \implies n \geq 12$$

5. Para qualquer $n \geq 12$, temos $\frac{3}{n} \leq \frac{1}{4} \implies -\frac{3}{n} \geq -\frac{1}{4}$. Subtraindo de $\frac{1}{2}$:

$$\frac{1}{2} - \frac{3}{n} \geq \frac{1}{2} - \frac{1}{4} = \frac{1}{4}$$

6. Multiplicando de volta por n^2 , obtemos:

$$\frac{1}{2}n^2 - 3n \geq \frac{1}{4}n^2, \quad \forall n \geq 12$$

Portanto, a definição de Ω é satisfeita com a constante $c = \frac{1}{4}$ e $n_0 = 12$.

e) **Prove que para qualquer constante** $k > 0$, $(n + k)^3 = \Theta(n^3)$.

Demonstração/Justificativa:

Precisamos encontrar $c_1, c_2 > 0$ e $n_0 \geq 0$ tais que $c_1 n^3 \leq (n + k)^3 \leq c_2 n^3$ para todo $n \geq n_0$.

• **Limite Inferior (c_1):**

Como $k > 0$, temos $n + k > n$. Elevando ao cubo em ambos os lados:

$$(n + k)^3 > n^3 = 1 \cdot n^3, \quad \forall n \geq 1$$

Logo, a constante $c_1 = 1$ funciona para $n_0 = 1$.

- **Limite Superior (c_2):**

Para todo $n \geq k$, vale que $k \leq n$. Logo, $n + k \leq n + n = 2n$. Elevando ao cubo:

$$(n + k)^3 \leq (2n)^3 = 8n^3, \quad \forall n \geq \lceil k \rceil$$

Logo, a constante $c_2 = 8$ funciona para $n_0 = \lceil k \rceil$.

Portanto, a relação $(n + k)^3 = \Theta(n^3)$ está provada com $c_1 = 1$, $c_2 = 8$ e $n_0 = \max(1, \lceil k \rceil) = \lceil k \rceil$.

5. Indique se as seguintes sentenças são verdadeiras ou falsas e justifique formalmente (com uma demonstração simples ou um contraexemplo):

a) $2^{n+1} = O(2^n)$.

Resposta: Verdadeiro.

Demonstração/Justificativa:

Pela propriedade de potências, reescrevemos o lado esquerdo como:

$$2^{n+1} = 2^1 \cdot 2^n = 2 \cdot 2^n$$

Pela definição da notação Big- O , queremos encontrar constantes $c > 0$ e $n_0 \geq 0$ tais que:

$$2 \cdot 2^n \leq c \cdot 2^n, \quad \forall n \geq n_0$$

Dividindo ambos os lados por 2^n (já que $2^n > 0$ para todo n):

$$2 \leq c$$

Portanto, basta escolher a constante $c = 2$ e $n_0 = 0$. Como a desigualdade $2^{n+1} \leq 2 \cdot 2^n$ se mantém verdadeira para todo $n \geq 0$, a afirmação é verdadeira.

b) $2^{2n} = O(2^n)$.

Resposta: Falso.

Demonstração/Justificativa:

Demonstração por Contradição:

Suponha por absurdo que $2^{2n} = O(2^n)$. Pela definição de Big- O , existiriam constantes $c > 0$ e $n_0 \geq 0$ tais que:

$$2^{2n} \leq c \cdot 2^n, \quad \forall n \geq n_0$$

Dividindo ambos os lados por 2^n :

$$\frac{2^{2n}}{2^n} \leq c \implies 2^n \leq c, \quad \forall n \geq n_0$$

Como c é uma constante fixa e 2^n cresce arbitrariamente ao infinito quando $n \rightarrow \infty$, é impossível que $2^n \leq c$ para todo $n \geq n_0$ (basta tomar $n > \log_2 c$). Chegamos a uma contradição. Logo, $2^{2n} \notin O(2^n)$.

c) **Se $f(n) = O(g(n))$, então $\log(f(n)) = O(\log(g(n)))$. (Assuma $\log(g(n)) \geq 1$).**

Resposta: Verdadeiro.

Demonstração/Justificativa:

Como $f(n) = O(g(n))$, existem constantes $c_0 > 0$ e $n_1 \geq 0$ tais que:

$$f(n) \leq c_0 \cdot g(n), \quad \forall n \geq n_1$$

Aplicando a função logarítmica (que é estritamente crescente) em ambos os lados:

$$\log(f(n)) \leq \log(c_0 \cdot g(n)) = \log(c_0) + \log(g(n))$$

Como a hipótese garante que $\log(g(n)) \geq 1$, podemos relacionar a constante $\log(c_0)$ com $\log(g(n))$:

- Se $c_0 \leq 1$, então $\log(c_0) \leq 0$, de onde segue que $\log(c_0) + \log(g(n)) \leq \log(g(n))$.
- Se $c_0 > 1$, então $\log(c_0) > 0$. Como $\log(g(n)) \geq 1$, temos que $\log(c_0) \leq \log(c_0) \cdot \log(g(n))$.

Unificando ambos os casos, obtemos:

$$\log(f(n)) \leq \log(c_0) + \log(g(n)) \leq |\log(c_0)| \cdot \log(g(n)) + 1 \cdot \log(g(n))$$

$$\log(f(n)) \leq (1 + |\log(c_0)|) \cdot \log(g(n))$$

Definindo a constante $c = 1 + |\log(c_0)|$ e $n_0 = n_1$, provamos formalmente que $\log(f(n)) = O(\log(g(n)))$.

d) $(n+1)! = O(n!)$.

Resposta: Falso.

Demonstração/Justificativa:**Demonstração por Teste do Limite:**

Pela definição de fatorial, sabemos que $(n+1)! = (n+1) \cdot n!$. Calculando o limite da razão entre as duas funções quando $n \rightarrow \infty$:

$$L = \lim_{n \rightarrow \infty} \frac{(n+1)!}{n!} = \lim_{n \rightarrow \infty} \frac{(n+1) \cdot n!}{n!} = \lim_{n \rightarrow \infty} (n+1) = \infty$$

Como $L = \infty$, a função $(n+1)!$ cresce assintoticamente mais rápido do que qualquer múltiplo constante de $n!$. Portanto, $(n+1)! \notin O(n!)$.

e) $\log^3 n = O(\sqrt{n})$.

Resposta: Verdadeiro.

Demonstração/Justificativa:**Demonstração por Mudança de Variável e Limite:**

Queremos avaliar a taxa de crescimento relativo via limite $L = \lim_{n \rightarrow \infty} \frac{\log^3 n}{\sqrt{n}}$.

Fazendo a mudança de variável $n = 2^{2k}$ (o que implica $\sqrt{n} = 2^k$ e $\log_2 n = 2k$), quando $n \rightarrow \infty$ temos $k \rightarrow \infty$:

$$L = \lim_{k \rightarrow \infty} \frac{(2k)^3}{2^k} = \lim_{k \rightarrow \infty} \frac{8k^3}{2^k}$$

Aplicando a regra de L'Hôpital sucessivamente 3 vezes para eliminar o polinômio do numerador:

$$L = \lim_{k \rightarrow \infty} \frac{24k^2}{2^k \ln 2} = \lim_{k \rightarrow \infty} \frac{48k}{2^k (\ln 2)^2} = \lim_{k \rightarrow \infty} \frac{48}{2^k (\ln 2)^3} = 0$$

Como o limite é igual a 0, isso prova não apenas que $\log^3 n = O(\sqrt{n})$, mas que de fato $\log^3 n = o(\sqrt{n})$ (a função polinomial cresce estritamente mais rápido do que a função polilogarítmica).

f) Se $f(n) = O(g(n))$ e $g(n) = O(h(n))$, então $f(n) = O(h(n))$.

Resposta: Verdadeiro.

Demonstração/Justificativa:

Demonstração da Transitividade da Notação O :

1. Como $f(n) = O(g(n))$, existem constantes $c_1 > 0$ e $n_1 \geq 0$ tais que:

$$f(n) \leq c_1 \cdot g(n), \quad \forall n \geq n_1$$

2. Como $g(n) = O(h(n))$, existem constantes $c_2 > 0$ e $n_2 \geq 0$ tais que:

$$g(n) \leq c_2 \cdot h(n), \quad \forall n \geq n_2$$

3. Escolhendo $n_0 = \max(n_1, n_2)$, ambas as desigualdades valem simultaneamente para todo $n \geq n_0$. Substituindo a segunda desigualdade na primeira:

$$f(n) \leq c_1 \cdot g(n) \leq c_1 \cdot (c_2 \cdot h(n)) = (c_1 \cdot c_2) \cdot h(n), \quad \forall n \geq n_0$$

Definindo a constante $c = c_1 \cdot c_2 > 0$, provamos que $f(n) \leq c \cdot h(n)$ para todo $n \geq n_0$, confirmando a propriedade transitiva $f(n) = O(h(n))$.

g) $f(n) + o(f(n)) = \Theta(f(n))$.

Resposta: Verdadeiro.

Demonstração/Justificativa:

Seja $h(n) = o(f(n))$. Pela definição formal da notação Little- o , para qualquer constante $\epsilon > 0$, existe um número $n_0 \geq 0$ tal que:

$$|h(n)| < \epsilon \cdot f(n), \quad \forall n \geq n_0$$

Escolhendo especificamente a constante $\epsilon = \frac{1}{2}$, existe um n_0 a partir do qual:

$$-\frac{1}{2}f(n) < h(n) < \frac{1}{2}f(n), \quad \forall n \geq n_0$$

Somando $f(n)$ em todos os membros da desigualdade:

$$f(n) - \frac{1}{2}f(n) < f(n) + h(n) < f(n) + \frac{1}{2}f(n)$$

$$\frac{1}{2}f(n) < f(n) + h(n) < \frac{3}{2}f(n), \quad \forall n \geq n_0$$

Como a desigualdade estrita $<$ implica na desigualdade não estrita $\leq$, temos:

$$\frac{1}{2}f(n) \leq f(n) + h(n) \leq \frac{3}{2}f(n), \quad \forall n \geq n_0$$

Pela definição de Θ , escolhendo as constantes $c_1 = \frac{1}{2}$ e $c_2 = \frac{3}{2}$, a relação $c_1 f(n) \leq f(n) + o(f(n)) \leq c_2 f(n)$ está satisfeita, provando que $f(n) + o(f(n)) = \Theta(f(n))$.

6. Prove ou refute as seguintes afirmações sobre transitividade assintótica:

Fundamentação Teórica (Definições Formais de O e o):

Sejam $f(n)$, $g(n)$ e $h(n)$ funções assintoticamente positivas definidas nos inteiros positivos.

- **Notação Big-O (O):** $f(n) \in O(g(n))$ se e somente se existem constantes reais $c > 0$ e $n_0 \geq 0$ tais que:

$$0 \leq f(n) \leq c \cdot g(n), \quad \forall n \geq n_0$$

- **Notação Little-o (o):** $f(n) \in o(g(n))$ se e somente se para **toda** constante real $\epsilon > 0$, existe uma constante $n_0 \geq 0$ tal que:

$$0 \leq f(n) < \epsilon \cdot g(n), \quad \forall n \geq n_0$$

Caracterização equivalente via limite: $f(n) \in o(g(n)) \iff \lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} = 0$.

- a) Se $f(n) \in O(g(n))$ e $g(n) \in o(h(n))$, então $f(n) \in o(h(n))$.

Resposta: Verdadeiro.

Demonstração/Justificativa:

Demonstração Formal por Definição (ϵ - n_0):

1. Pela hipótese $f(n) \in O(g(n))$, existem constantes $c_1 > 0$ e $n_1 \geq 0$ tais que:

$$f(n) \leq c_1 \cdot g(n), \quad \forall n \geq n_1$$

2. Pela hipótese $g(n) \in o(h(n))$, sabemos que para **qualquer** $\epsilon' > 0$, existe $n_2 \geq 0$ tal que:

$$g(n) < \epsilon' \cdot h(n), \quad \forall n \geq n_2$$

3. Queremos provar que $f(n) \in o(h(n))$, ou seja, que dado um $\epsilon > 0$ arbitrário, existe um $n_0 \geq 0$ tal que $f(n) < \epsilon \cdot h(n)$ para todo $n \geq n_0$.
4. Dado $\epsilon > 0$, escolha especificamente $\epsilon' = \frac{\epsilon}{c_1} > 0$. Para esta escolha de ϵ' , a hipótese (2) garante a existência de um n_2 adequado.
5. Definindo $n_0 = \max(n_1, n_2)$, temos que para todo $n \geq n_0$, ambas as desigualdades se aplicam simultaneamente:

$$f(n) \leq c_1 \cdot g(n) < c_1 \cdot \left(\frac{\epsilon}{c_1} \cdot h(n) \right) = \epsilon \cdot h(n), \quad \forall n \geq n_0$$

Como mostramos que $f(n) < \epsilon \cdot h(n)$ para qualquer $\epsilon > 0$ quando $n \geq n_0$, provamos formalmente que $f(n) \in o(h(n))$.

- b) Se $f(n) \in o(g(n))$ e $g(n) \in O(h(n))$, então $f(n) \in o(h(n))$.

Resposta: Verdadeiro.

Demonstração/Justificativa:

Demonstração Formal por Definição (ϵ - n_0):

1. Pela hipótese $g(n) \in O(h(n))$, existem constantes $c_2 > 0$ e $n_2 \geq 0$ tais que:

$$g(n) \leq c_2 \cdot h(n), \quad \forall n \geq n_2$$

2. Pela hipótese $f(n) \in o(g(n))$, sabemos que para **qualquer** $\epsilon' > 0$, existe $n_1 \geq 0$ tal que:

$$f(n) < \epsilon' \cdot g(n), \quad \forall n \geq n_1$$

3. Queremos provar que $f(n) \in o(h(n))$, ou seja, dado um $\epsilon > 0$ qualquer, encontrar n_0 tal que $f(n) < \epsilon \cdot h(n)$ para todo $n \geq n_0$.

4. Dado $\epsilon > 0$, escolha especificamente $\epsilon' = \frac{\epsilon}{c_2} > 0$. Para esta escolha de ϵ' , a hipótese (2) nos dá um n_1 adequado.
5. Definindo $n_0 = \max(n_1, n_2)$, aplicamos as duas desigualdades para todo $n \geq n_0$:

$$f(n) < \epsilon' \cdot g(n) = \left(\frac{\epsilon}{c_2}\right) \cdot g(n) \leq \left(\frac{\epsilon}{c_2}\right) \cdot (c_2 \cdot h(n)) = \epsilon \cdot h(n), \quad \forall n \geq n_0$$

Como $f(n) < \epsilon \cdot h(n)$ para todo $n \geq n_0$, concluimos que $f(n) \in o(h(n))$.