UNIVERSIDADE DE BRASÍLIA
FACULDADE DO GAMA

CURSO:	**ENGENHARIAS**		
DISCIPLINA:	**Estruturas de Dados e Algoritmos**	**CÓDIGO:**	**193704**
CARGA HORÁRIA:	**60 h**	**CRÉDITOS:**	**04**
PROFESSOR:	**Dr. Nilton Correia da Silva e Dr. Fabricio Ataides Braz**		

RESOLUÇÃO DE TRABALHO PRÁTICO
TEMA: NOTAÇÃO ASSINTÓTICA

1. Para os pares de funções abaixo, determine o menor valor de n ($n \in N$) para o qual a segunda função (g(n)) se torna menor do que a primeira (f(n)). Isto é, a partir de que valor inteiro positivo f(n) domina assintoticamente g(n) nos itens logo a seguir?:
   a. $f(n) = n^2, g(n) = 10n$

   $f(n) = g(n) \rightarrow n^2 = 10n \rightarrow \boldsymbol{n = 10}$

   b. $f(n) = n.\log n,\ g(n) = 2n$

   $f(n) = \ g(n) \rightarrow n.logn = 2n \rightarrow logn = 2 \rightarrow 2^2 = n \rightarrow \boldsymbol{n = 4}$

2. Escrever as seguintes funções em notação O:
   a. $f(n) = n^3 - 1,$ $f(n) = O(n^3)$
   b. $f(n) = n^2 + \log n,$ $f(n) = O(n^2)$
   c. $f(n) = 3.n^n + 5.2^n,$ $f(n) = O(n^n)$
   d. $f(n) = (n-1)^n + n^{(n-1)},$ $f(n) = O(n^n)$
   e. $f(n) = 345$ $f(n) = O(345)$

3. Indique se as afirmativas abaixo são verdadeiras ou falsas e justifique a sua resposta.
   a. $2^{n+1} = O(2^n),$ Falso.
   2^{n+1} cresce a uma taxa menor ou igual a $2^k, \forall k \geq n+1$: $2^{n+1} = O(2^{n+1})$

   b. $2^{2n} = O(2^n),$ Falso.
   $2^{2n} = 4^n$. Logo: $2^{2n} = O(4^n)$

   c. $3^{n+k} = \Omega(3^n), \forall k > 0$ Verdadeiro.
   $3^{n+k} = \Omega(3^{n+k})$, pois para qualquer $k > 0 \rightarrow 3^{n+k} > 3^n$

   d. $f(n) = n^3 + n^2 + 4, g(n) = 3.n^3 + 2 \rightarrow f(n) = \Theta(g(n))$ Verdadeiro.
      1. $f(n) = O(n^3)$ e $f(n) = \Omega(n^3)$, logo $f(n) = \Theta(n^3)$;
      2. $g(n) = O(n^3)$ e $g(n) = \Omega(n^3)$, logo $g(n) = \Theta(n^3)$;
      3. Concluímos que $f(n) = \Theta(g(n))$;

e. $f(n) = O\big(u(n)\big)\ e\ g(n) = O\big(v(n)\big) \ \rightarrow f(n) + g(n) = O\big(u(n) + v(n)\big)$ Verdadeiro
Propriedade da soma.

f. $f(n) = O\big(u(n)\big)\ e\ g(n) = O\big(v(n)\big) \ \rightarrow f(n) + g(n) = O\left(\max\left(u(n), v(n)\right)\right)$ Verdadeiro
Propriedade da soma, pois $O\left(\max\left(u(n), v(n)\right)\right) = O\big(u(n) + v(n)\big)$.

g. $f(n) = O\big(u(n)\big)\ e\ g(n) = O\big(v(n)\big) \ \rightarrow f(n).g(n) = O\big(u(n).v(n)\big)$ Verdadeiro
Propriedade da multiplicação.

4. Considere um algoritmo de complexidade $\Theta(n^4)$. Para n suficientemente grande, o algoritmo demorou um tempo t. Estime o tempo gasto se triplicarmos o tamanho da instância (n).
Para uma instância de n, tem-se que o tempo demandado pelo algoritmo é $t = n^4$;
Triplicando a instância, o que acontece com o tempo demandado?
$t = (3.n)^4 \rightarrow t = 3^4.n^4 \rightarrow \boldsymbol{t = 81n^4}$

5. Considere um algoritmo de complexidade $\Theta(\log n)$. Para n suficientemente grande, o algoritmo demorou um tempo t. Estime o tamanho da estância (n) que pode ser processada no tempo 2t.
Para uma instância de n, tem-se que o tempo demandado pelo algoritmo é $t = \log n$;
Duplicando o tempo disponível, pode-se processar que tamanho de instância?
$2t = \log n \rightarrow logn = 2t \rightarrow \boldsymbol{n = 2^{2t}}$.