FUNDAÇÃO GETÚLIO VARGAS

ESCOLA DE MATEMÁTICA APLICADA

MESTRADO 2015.1

ESTRUTURA DE DADOS E SEUS ALGORITMOS

Prof Alexandre Rademaker

Resolução dos Exercícios Selecionandos do Capítulo 0

GRUPO:

KIZZY TERRA

GUSTAVO AVILA

CAROLINE FIALHO

RIO DE JANEIRO

MARÇO DE 2015

1 Exercício 0.1

Obs: Os gráficos aqui apresentados foram desenhados utilizando-se o WolframAlpha disponível em: http://www.wolframalpha.com.

a) $f(n)$= n-100 **,** $g(n)$= n-200

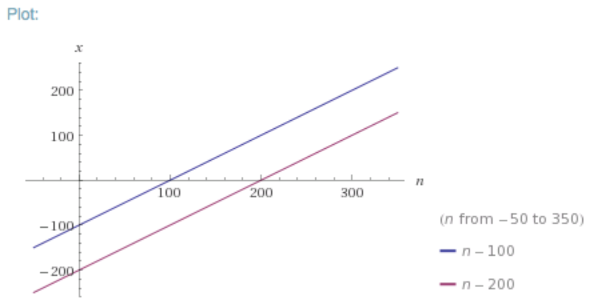

Figure 1: $f(n)$= n-100 , $g(n)$= n-200

A partir do gráfico podemos observar que o crescimento das funções é o mesmo, diferenciando-se apenas por um deslocamento vertical. Dessa forma, f = Θ(g).

b) $f(n)$= $n^{1/2}$ **,** $g(n)$= $n^{2/3}$

Da regra 2 apresentada na seção 0.3 do capítulo 0, sabemos que n^a domina n^b se a > b. Logo f = O(g)

c) $f(n) = 100n + \log n$**,** $g(n) = n + (\log n)^2$

Da regra 4 apresentada na seção 0.3 do capítulo 0, sabemos que qualquer polinômio domina qualquer logaritmo. Assim, $f(n) = O(n)$ e $g(n) = O(n)$ e portanto f = Θ(g).

d) $f(n) = n \log n$**,** $g(n)$ =$10n \log n$

Da regra 1 apresentada na seção 0.3 do capítulo 0, sabemos que constantes multiplicativas podem ser omitidas. Assim, $f(n) = n\log n$ e $g(n) = O(n\log n)$ e portanto f = Θ(g).

e) $f(n) = \log 2n$**,** $g(n)$ =$\log 3n$

Analogamente ao item anterior, pela regra 1, as constantes multiplicativas podem ser omitidas. Assim, $f(n) = O(\log n)$ e $g(n) = O(\log n)$ e portanto f = Θ(g).

f) $f(n) = 10 \log n$**,** $g(n)$ =$\log(n^2)$

Analisando o gráfico abaixo, podemos perceber que f(n) cresce mais rápido do que g(n) portanto f = Ω(g).

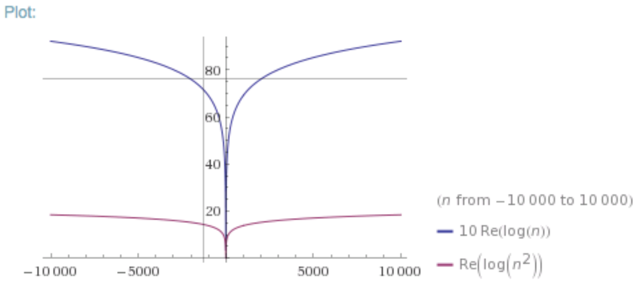

Figure 2: $f(n) = 10 \log n$, $g(n)$ =log(n^2)

g) $f(n)$= $n^{1.01}$**,** $g(n) = nlog^2 n$

Analisando o gráfico abaixo, podemos perceber que f(n) cresce mais rápido do que g(n) portanto f = Ω(g).

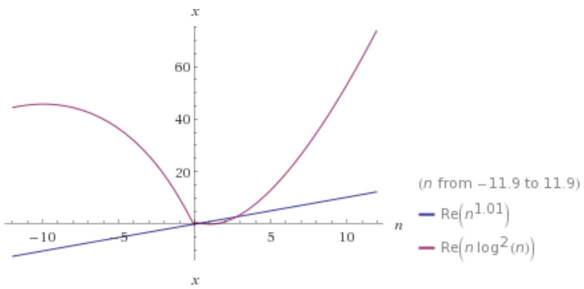

Figure 3: $f(n)$= $n^{1.01}$, $g(n) = nlog^2 n$

h) $f(n) = n^2/\log n$**,** $g(n) = n(\log n)^2$

Utilizando a regra 4 apresentada na seção 0.3 do capítulo 0, podemos afirmar que $f(n) = O(n^2)$ e $g(n) = O(n)$. Decorre disso que f domina g e portanto f = Ω(g).

No gráfico abaixo pode se observar facilmente que f domina g.

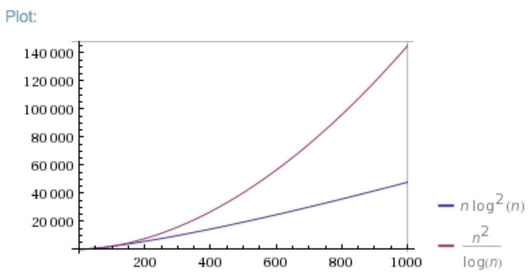

Figure 4: $f(n) = n^2/\log n$, $g(n) = n(\log n)^2$

i) $f(n) = n^{0.1}$, $g(n) = (\log n)^{10}$

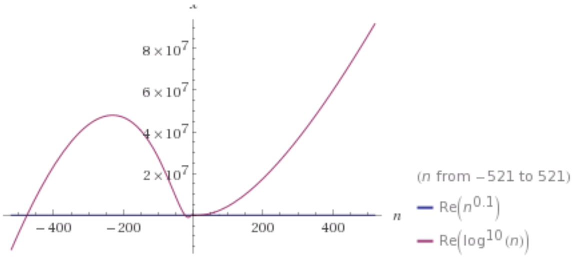

Figure 5: $f(n) = n^{0.1}$, $g(n) = (\log n)^{10}$

O gráfico acima evidencia que g domina f, logo f = O(g).

j) $f(n) = (\log n)^{\log n}$, $g(n) = n/\log n$

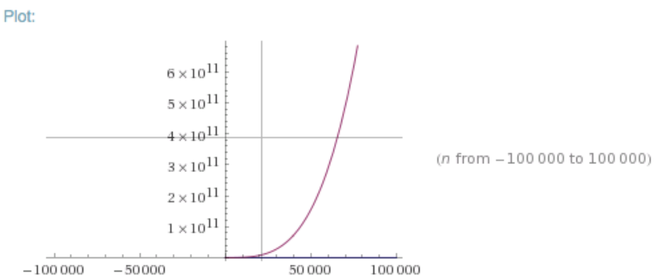

Figure 6: $f(n)$ (em vermelho) , $g(n)$ (em azul)

O gráfico acima evidencia que f domina g, logo f = Ω(g).

k) $f(n) = \sqrt{n}$, $g(n) = (\log n)^3$

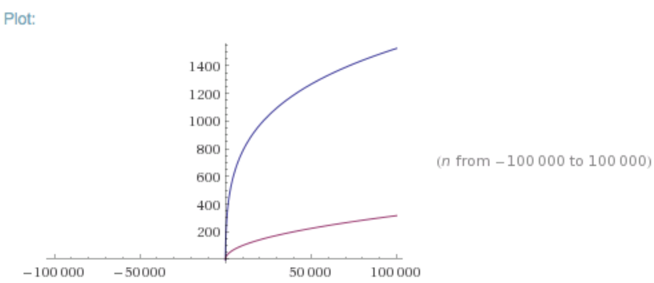

Figure 7: $f(n)$ (em vermelho) , $g(n)$ (em azul)

O gráfico acima evidencia que g domina f, logo f = O(g).

l) $f(n)= n^{1/2}$ **,** $g(n) = 5^{\log_2 n}$

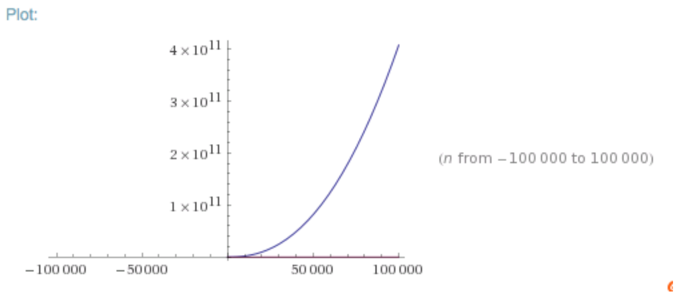

Figure 8: $f(n)$ (em vermelho) , $g(n)$ (em azul)

O gráfico acima evidencia que g domina f, logo f = O(g).

m) $f(n)= n2^n$**,** $g(n) = 3^n$

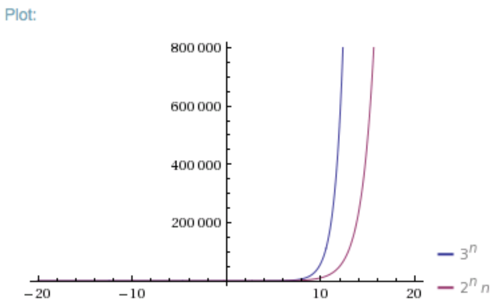

Figure 9: $f(n)= n2^n$, $g(n) = 3^n$

O gráfico acima evidencia que as duas funções crescem a mesma taxa e portanto, f = Θ(g).

n) $f(n)= 2^n$**,** $g(n) = 2^{n+1}$

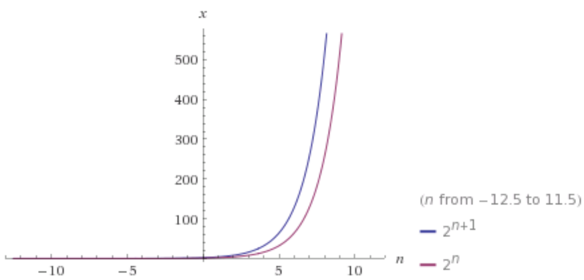

Figure 10: $f(n)= 2^n$, $g(n) = 2^{n+1}$

O gráfico acima evidencia que as duas funções crescem a mesma taxa e portanto, f = Θ(g).

o) $f(n)$= $n!$**,** $g(n) = 2^n$

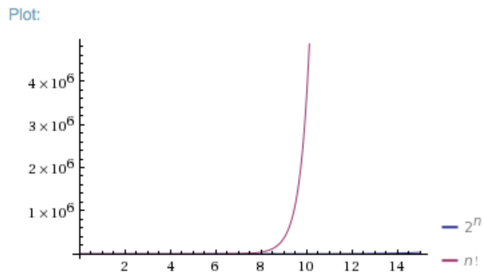

Figure 11: $f(n)$= $n!$, $g(n) = 2^n$

O gráfico acima evidencia que f domina g, logo f = Ω(g).

p) $f(n) = (\log n)^{\log n}$**,** $g(n) = 2^{(\log_2 n)^2}$

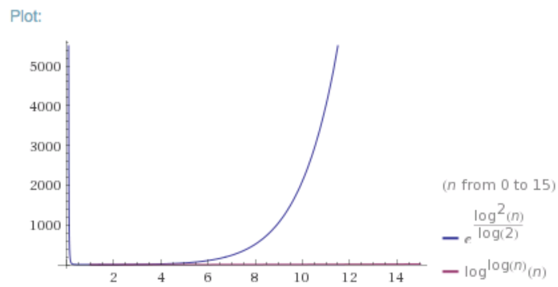

Figure 12: $f(n) = (\log n)^{\log n}$, $g(n) = 2^{(\log_2 n)^2}$

O gráfico acima evidencia que g domina f, logo f = O(g).

q) $f(n) = \sum_{i=1}^{n} i^k$**,** $g(n) = n^{k+1}$

Temos que $f(n) = 1^k + 2^k + ... + n^k$ e $g(n) = n^{k+1}$.

É possível encontrar um polinômio de grau k+1 em n que equivale a $f(n)$. Basta resolver um sistema de k+1 equações para encontrar os k+1 coeficientes correspondentes do polinômio, este sistema tem solução e ela é única. Assim, se $f(n) = O(n^{k+1})$ e $g(n) = n^{k+1}$ temos que f = Θ(g).

2 Exercício 0.2

a) $\Theta(1)$ se c < 1

Tendo em vista que c < 1 e considerando a função $g(n)$= 1+c+c^2+...+c^n como uma progressão geométrica de razão c (para c real positivo, diferente de 1) cuja soma é dada por:

$$\frac{c^{n+1}-1}{c-1}$$

Se fizermos o limite da função g(n) quando n tende a infinito teremos:

$$\lim\inf(\tfrac{c^{n+1}-1}{c-1}) = \frac{0-1}{c-1} = \frac{1}{1-c}$$

Isto é, o valor da série será igual a uma constante que iremos chamar de p. Assim $g(n) = p$, iremos provar que g(n) = $\Theta(1)$.

Seja a função $f(n)$= 1e seja $p + 1 > p$, teremos:

1) $g = O(f)$

$$g(n) \leq (p+1)f(n) \Rightarrow \frac{g(n)}{f(n)} \leq (p+1) \Rightarrow g = O(f)$$

2) $f = O(g)$

$$f(n) \leq g(n) \Rightarrow \frac{f(n)}{g(n)} \leq 1 \Rightarrow f = O(g)$$

Portanto,

$$g = \Theta(f) \Rightarrow g = \Theta(1)$$

b) $\Theta(n)$ se c = 1

Se c = 1, temos:

$$g(n) = 1 + c + c^2 + ... + c^n = 1 + 1 + 1^2 + ... + 1^n = n + 1$$

Iremos provar que g(n) = Θ(n):

Considere a função $f(n)= n$, queremos:

1) $g = O(f)$

$$g(n) \leq 2 * f(n) \Rightarrow \frac{g(n)}{f(n)} \leq 2 \Rightarrow g = O(f)$$

2) $f = O(g)$

$$f(n) \leq g(n) \Rightarrow \frac{f(n)}{g(n)} \leq 1 \Rightarrow f = O(g)$$

Portanto,

$$g = \Theta(f) \Rightarrow g = \Theta(n)$$

c) $\Theta(c^n)$ se c < 1

Se c > 1, teremos que$g(n) = 1 + c + c^2 + ... + c^n$é uma progressão geométrica de razão c cuja soma é dada por:

$$\frac{c^{n+1}-1}{c-1}$$

Iremos provar que g(n) = $\Theta(c^n)$:

Considere a função $f(n) = c^n$, queremos:

1) $g = O(f)$

$$\frac{c^{n+1}-1}{c-1} \leq c * c^n = c^{n+1}, \Rightarrow g(n) \leq c * f(n) \Rightarrow \frac{g(n)}{f(n)} \leq c \Rightarrow g = O(f)$$

2) $f = O(g)$

$$c^n \leq (c-1) * \frac{c^{n+1}-1}{c-1} \Rightarrow f(n) \leq c * g(n) \Rightarrow \frac{f(n)}{g(n)} \leq 1 \Rightarrow f = O(g)$$

Portanto,

$$g = \Theta(f) \Rightarrow g = \Theta(c^n)$$

3 Exercício 0.3

a) Prova por indução:

1) Para n = 6, $F_6 = F_5 + F_4 = 8$
2) Para n = 7, $F_7 = F_6 + F_5 = 13$
3) Supondo que $F_n \geq 2^{0.5n}$ para $n \geq 8$, devemos mostrar que também é válido para $n+1$:

$$F_n + F_{n-1} \geq 2^{0.5n} + 2^{0.5(n-1)} = 2^{n/2} + 2^{(n-1)/2} = 2^{(n-1)/2}.(2^{1/2} + 1) \geq 2^{(n-1)/2}.2 = 2^{(n+1)/2}$$

$$\Rightarrow F_{n+1} \geq 2^{0.5(n+1)}$$

b) Para resolver este item utilizamos como ferramenta auxiliar o Microsoft Excel, calculando valores para F_ne 2^{cn} e traçando seus gráficos para analisar a tendência das curvas e verificar para que valores $F_n \leq 2^{cn}$.

Primeiramente escolhemos c= 0.9 e verificamos o crescimento das curvas. O gráfico abaixo evidencia que a curva 2^{cn} cresce muito mais rápido do que a curva $F_n$para c = 0.9.

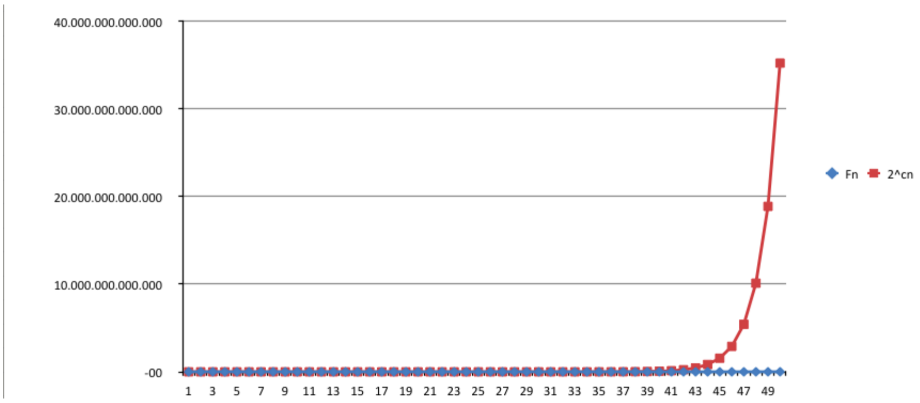

Figure 13: F_ne 2^{cn}

c) Para resolver este item utilizamos o mesmo método do item anterior (análise de gráfico) e fomos escolhendo valores para c, de forma que as curvas se aproximassem suficientemente e mantivessem a relação $F_n \leq 2^{cn}$.

O valor limite encontrado para c foi aproximadamente 0,665 cujo gráfico correspondente apresenta-se a seguir:

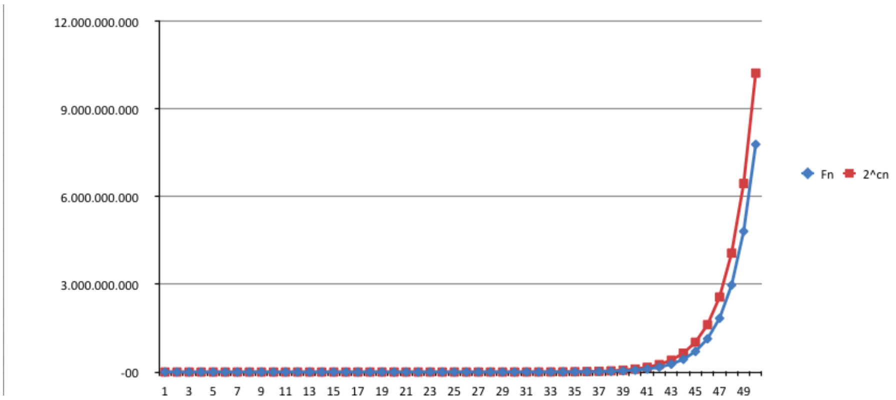

Figure 14: F_ne 2^{cn}