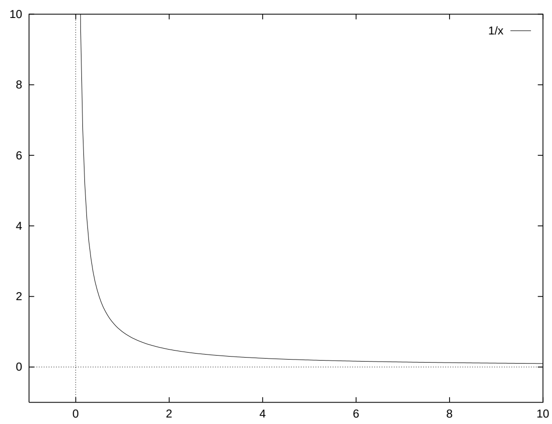
Notação assintótica
Prof. João B. Oliveira

Princípios

Quando os matemáticos gregos estudaram as seções cônicas (elipse, parábola e hipérbole), encontraram situações como a representada abaixo pela hipérbole $y = 1/x$. Neste caso, a curva aproxima-se cada vez mais dos dois eixos x e y, porém sem nunca tocá-los. Desde então, cunhou-se a palavra "assíntota" para denotar este tipo de retas, às quais uma função se aproxima cada vez mais porém não atinge.



Atualmente, usamos a expressão "assintótico" em um sentido mais amplo, para falar de algo que se aproxima cada vez mais de um certo valor, à medida que um parâmetro aproxima-se de um limite (geralmente $\pm\infty$). Podemos dizer que para nós "assintótico" significa "aproximando-se cada vez mais".

A hierarquia básica

Na prática, funções de uma variável n crescem e diminuem de formas diferentes, com algumas aproximando-se ou afastando-se de ∞ (ou outros valores "interessantes") mais rápido do que outras. Para formalizar a idéia, podemos introduzir a relação de crescimento $\prec$:

$$f(n) \prec g(n) \Longleftrightarrow \lim_{n\to\infty} \frac{f(n)}{g(n)} = 0.$$

Desta forma, expressamos o fato de que "$f(n)$ cresce mais devagar do que $g(n)$ à medida que n cresce". Esta relação é transitiva, ou seja

$$f(n) \prec g(n),\ g(n) \prec h(n) \Longleftrightarrow f(n) \prec h(n).$$

Também podemos escrever de modo inverso, ou seja, escrever $h(n) \succ f(n)$. Por exemplo, sabemos que $n \prec n^2$, e informalmente dizemos que n cresce mais lentamente do que n^2. De forma mais geral, sabemos que

$$n^\alpha \prec n^\beta \Longleftrightarrow \alpha < \beta,$$

onde α e β são números reais e $1 \leq \alpha$. Podemos usar $\prec$ para classificar muitas (mas não todas!) funções em uma ordem assintótica, incluindo por exemplo as funções abaixo:

$$1 \prec \log\log n \prec \log n \prec n^c \prec c^n.$$

Neste exemplo, c é uma constante qualquer maior do que um, e todas as funções listadas (exceto a primeira, naturalmente) vão ao infinito à medida que n vai ao infinito. Assim, quando inserimos uma nova função nesta hierarquia não estamos determinando *se* ela vai para o infinito, mas sim *com que velocidade*.

A hierarquia acima apresenta algumas funções que vão para o infinito. Entretanto, algumas vezes estamos interessados em funções que vão para zero à medida que n cresce. Para isso, é útil ter uma hierarquia similar. Supondo que $f(n)$ e $g(n)$ nunca são zero, podemos usar as recíprocas e obter

$$f(n) \prec g(n) \iff \frac{1}{g(n)} \prec \frac{1}{f(n)}.$$

Quando duas funções $f(n)$ e $g(n)$ tem *a mesma* taxa de crescimento, escrevemos $f(n) \asymp g(n)$, cuja definição formal é dada abaixo:

$$f(n) \asymp g(n) \iff |f(n)| \leq C|g(n)| \ \text{ e } \ |g(n)| \leq C|f(n)|.$$

Neste caso, C é considerada uma constante qualquer, a ser escolhida de forma a satisfazer as desigualdades.

Notação O

A notação hoje usada para análise assintótica foi introduzida por Bachmann no final do século passado e tem o objetivo de suprimir detalhes que não são importantes em fórmulas e salientar os fatores mais relevantes. Isto será demonstrado com um pequeno exemplo: os números harmônicos H_n são definidos da seguinte forma:

$$H_n = \frac{1}{1} + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{n} = \sum_{k=1}^{n} \frac{1}{k}.$$

Sendo assim, os primeiros números harmônicos são:

n	0	1	2	3	4	5	6	7	8	9	10
H_n	0	1	$\frac{3}{2}$	$\frac{11}{6}$	$\frac{25}{12}$	$\frac{137}{60}$	$\frac{49}{20}$	$\frac{363}{140}$	$\frac{761}{280}$	$\frac{7129}{2520}$	$\frac{7381}{2520}$

Avaliando os mesmos números de forma aproximada, temos:

n	0	1	2	3	4	5	6	7	8	9	10
H_n	0	1	1.5	1.8333	2.0833	2.2833	2.45	2.5928	2.7178	2.8289	2.9289

A pergunta que nos interessa é: à medida que n cresce, como crescem os números H_n? Em primeiro lugar, podemos usar um resultado do Cálculo para mostrar que $H_n \to \infty$ à medida que $n \to \infty$, mas queremos saber mais detalhes sobre o crescimento dos números H_n.

Dando um salto *muito* grande (e sem mostrar o desenvolvimento, que não nos interessa no momento), podemos apelar diretamente para um resultado que diz o seguinte:

$$H_n = \log_2 n + \gamma + \frac{1}{2n} - \frac{1}{12n^2} + \frac{\epsilon}{120n^4}, \qquad 0 < \epsilon < 1.$$

É importante notar o papel de ϵ, que serve como avaliação do erro cometido pelo resto da expressão. (Na fórmula, γ é a constante de Euler, e vale aproximadamente 0.5772).

Com a fórmula acima, o valor de H_n pode ser calculado diretamente, sem somar as frações uma a uma, cometendo-se um erro de no máximo $1/120n^4$. Como este último termo não pode ser avaliado com precisão, a fórmula anterior passa a ser representada assim:

$$H_n = \log_2 n + \gamma + \frac{1}{2n} - \frac{1}{12n^2} + O\left(\frac{1}{120n^4}\right). \tag{1}$$

Desta forma, estamos representando a *ordem de grandeza* do último termo, dizendo que a soma deve ser feita e ao final pode ser que se precise adicionar um valor que é um múltiplo de $1/120n^4$. Vale dizer que se nosso resultado inicial não fosse tão preciso, poderíamos dizer algo diferente. O exemplo abaixo também seria válido:

$$H_n = \log_2 n + \gamma + O\left(\frac{1}{2n}\right). \tag{2}$$

Neste caso, como temos menos informação disponível, sabemos apenas que o valor a ser adicionado ao final é de no máximo alguma constante vezes $1/2n$, ou seja, bem maior do que $1/120n^4$ se n for grande.

Como mostrado, a notação O é usada para transmitir a noção de ordem de grandeza, mas poupando-nos de detalhes no tratamento dos termos de menor importância em expressões[1].

Definição: notação O

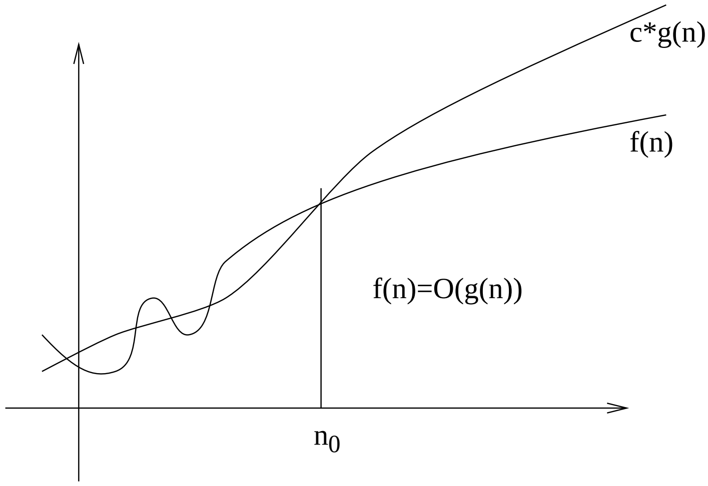
Até o momento, a notação O foi apresentada, mas não foi definida apropriadamente, contentando-nos com uma explicação simplificada de seu funcionamento. Agora, temos a oportunidade de apresentar a definição das notações assintóticas mais usuais, e rever o exemplo acima de acordo com elas.

A notação O serve para dar um *limite superior assintótico*, e para uma dada função $g(n)$ denotamos como $O(g(n))$ o conjunto de funções

$$O(g(n)) = \{f(n) \mid \begin{array}{l} \text{existem constantes positivas } c \text{ e } n_0 \text{ tais que} \\ 0 \leq f(n) \leq c \cdot g(n) \text{ para todo } n > n_0\}. \end{array}$$

Apesar da apresentação pouco intuitiva, a noção por trás da definição pode ser apresentada na figura a seguir:

[1] Mais tarde, veremos que até mesmo as expressões (1) e (2) anteriores são complicadas e podem ser reduzidas.



Partindo da definição, vemos que $O(g(n))$ não é uma única função, mas sim um conjunto de funções que, a partir de um certo valor n_0, são menores do que a função $g(n)$ multiplicada por alguma constante. Os seguintes detalhes devem ser lembrados:

- Dentro do conjunto $O(g(n))$ estão infinitas funções $f(n)$, e cada uma pode ter seu próprio valor para n_0 e para a constante c.
- Pode-se considerar $O(g(n))$ como uma função (como em uma linguagem de programação) que recebe um parâmetro $(g(n))$ e é avaliada de acordo, gerando um conjunto de funções.
- As funções $f(n)$ e $g(n)$ são obrigatoriamente positivas.

Desta forma, as funções que pertencem a $O(g(n))$ serão, a partir de certo ponto, menores do que $g(n)$ (se for preciso, esta última pode ter de ser multiplicada por uma constante.)

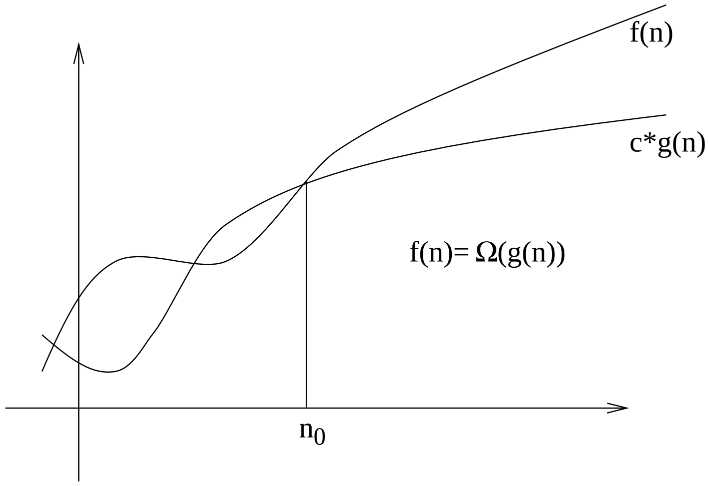
Além da notação O, existem outras duas notações que são importantes: uma tem o papel inverso de O e pode ser usada como *limite inferior assintótico* e a última é uma combinação das duas e serve como *limite assintótico estrito* para $f(n)$. Elas serão apresentadas a seguir:

Definição: notação Ω

A notação Ω funciona de maneira inversa a O, e limita inferiormente as funções definidas por ela. Denotamos como $\Omega(g(n))$ o conjunto de funções

$$\Omega(g(n)) = \{f(n) \mid \begin{array}{l} \text{existem constantes } c \text{ e } n_0 \text{ tais que} \\ 0 \leq c \cdot g(n) \leq f(n) \text{ para todo } n > n_0\}. \end{array}$$

A noção expressa nesta definição é mostrada mais claramente na figura a seguir:



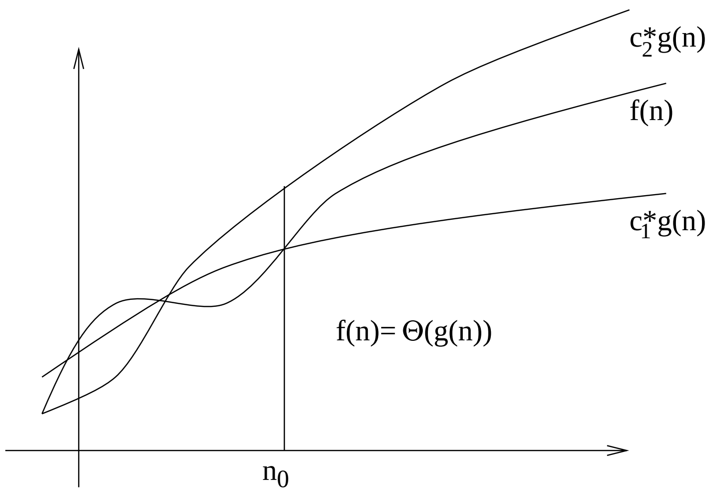
Desta forma, percebe-se a similaridade entre os conceitos das duas notações. Desta vez, o conjunto de funções $\Omega(g(n))$ compõe-se de todas as funções que, a partir de certo ponto, são maiores do que $g(n)$ multiplicada por uma constante.

Definição: notação Θ

A notação Θ pode ser vista como uma junção das notações O e Ω. Estas duas são limitantes de forma separada, e a notação Θ serve para limitar funções pelos dois lados, ou seja, superior e inferiormente. Sendo assim, a definição é uma composição das definições anteriores:

$$\Theta(g(n)) = \{f(n) \mid \begin{array}{l} \text{existem constantes } c_1,\ c_2 \text{ e } n_0 \text{ tais que} \\ 0 \leq c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n) \text{ para todo } n > n_0\}. \end{array}$$

Segundo a definição, uma função $f(n)$ pertence ao conjunto $\Theta(g(n))$ se existem constantes c_1 e c_2 que fazem com que ela seja prensada entre $c_1 \cdot g(n)$ e $c_2 \cdot g(n)$, para algum n suficientemente grande. A figura a seguir mostra esta situação.



O teorema abaixo serve para interligar as três notações acima, de uma forma que deve ser bastante intuitiva ao leitor:

Teorema: Para quaisquer duas funções $f(n)$ e $g(n)$, $f(n) = \Theta(g(n))$ se e somente se $f(n) = O(g(n))$ e $f(n) = \Omega(g(n))$.

Avaliando onde?

Uma pergunta que sempre é feita refere-se ao valor mais apropriado para avaliar as funções, para que se possa dizer quem é maior do que quem. Isto é bastante natural, pois dependendo de onde se olha, as funções podem comportar-se de maneira completamente diferente. Para solucionar a questão, adota-se a seguinte opção: sempre considera-se o comportamento das funções à medida que sua variável cresce em direção ao infinito, ou seja à medida que $n \to \infty$. Desta maneira, descartamos o comportamento das funções para valores específicos de n, preferindo pensar no que acontece quando a variável cresce cada vez mais. Esta é uma decisão importante e terá muitas consequ"ências para a avaliação de algoritmos.

Abordagem alternativa

Se você se sente melhor trabalhando com limites, no lugar das definições às vezes um pouco obscuras para O, Ω e Θ, você pode usar o quadro abaixo, que ajuda bastante a perceber o significado desta notação:

"f é...	significa que f cresce	e se escreve	e equivale a
... O de g"	não mais depressa que g	$f = O(g)$	$\exists n_0, c > 0 : \forall n > n_0, f(n) < c.g(n)$
... theta de g"	aproximadamente como g	$f = \Theta(g)$	$f = O(g)$ e $g = O(f)$
... como g"	da mesma forma que g	$f \approx g$	$\lim_{n\to\infty} \frac{f(n)}{g(n)} = 1$
... ômega de g"	não mais devagar que g	$f = \Omega(g)$	$g = O(f)$

Outra forma de exprimir as mesmas definições é através das seguintes regras de limites:

1. Se $\lim_{n\to\infty} \frac{f(n)}{g(n)} \in R^+$, então $f(n) = O(g(n))$ e $g(n) = O(f(n))$.
2. Se $\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0$, então $f(n) = O(g(n))$ mas $g(n) \notin O(f(n))$.
3. Se $\lim_{n\to\infty} \frac{f(n)}{g(n)} = +\infty$, então $f(n) \notin O(g(n))$ mas $g(n) = O(f(n))$.

Note como todos os conceitos estão amarrados à notação O. Agora, aproveite e faça o exercício de achar pares de funções f, g tais que

1. $f \approx g$
2. $f = \Theta(g)$
3. $f = \Omega(g)$

Manipulação

Antes de qualquer coisa, deve-se lembrar sempre que $O(g(n))$ é um *conjunto*, e deve ser tratado como tal. Ou seja: não é uma função, nem uma fórmula, nem nada parecido. É um *con-jun-to!* Desta maneira, as operações que podemos usar são extremamente bem definidas. No entanto, a tradição permite uma única quebra de protocolo: em geral, deveríamos escrever coisas como

$$\frac{n^3}{3} + \frac{n^2}{2} + \frac{n}{6} \in O(n^3), \tag{3}$$

significando que a expressão do lado esquerdo é limitada superiormente por n^3. No entanto geralmente encontraremos a mesma idéia expressa assim:

$$\frac{n^3}{3} + \frac{n^2}{2} + \frac{n}{6} = O(n^3). \tag{4}$$

Em princípio isto está *errado*, mas como geralmente pode-se reconhecer facilmente a situação, toma-se esta liberdade. Porém, vale salientar que as expressões com a notação O (ou Θ ou Ω) nunca são escritas *sozinhas* do lado esquerdo, ou seja, é completamente inaceitável escrever

$$O(n^3) = \frac{n^3}{3} + \frac{n^2}{2} + \frac{n}{6}, \tag{5}$$

porque neste caso já estaríamos abusando da notação, e poderíamos deduzir coisas absurdas a partir de fatos verdadeiros, como por exemplo

$$\begin{aligned} n^2 &= O(n^3) && (6)\\ O(n^3) &= n && (7)\\ &\Rightarrow n^2 = O(n^3) = n \Rightarrow n^2 = n && (8) \end{aligned}$$

Pode-se explicar o problema de outra forma, dizendo que ao escrevermos da forma (3) ou (4) estamos reduzindo a informação do lado esquerdo (onde há mais informação, com detalhes sobre a expressão) para o lado direito (muito menos detalhe, apenas uma aproximação). Já ao fazermos o contrário, da forma (5), estaríamos concluindo mais informação do que temos inicialmente, o que certamente causará problemas. Desta forma, enquanto (6) está correta, escrever (7) é inapropriado e permite chegar à conclusão (8), que evidentemente está errada.

No entanto, é válido escrever

$$\frac{n^3}{3} + O(n^2) = O(n^3),$$

pois isto quer dizer que um conjunto de funções (todas compostas de $n^3/3$ mais uma quantidade desconhecida, porém limitada superiormente por n^2) está contido em outro conjunto de funções limitado superiormente por n^3.

Propriedades

A seguir estão algumas propriedades das notações apresentadas anteriormente:

Propriedades básicas

$$
\begin{aligned}
f(n) &= O(f(n)) && (9)\\
c \cdot O(f(n)) &= O(f(n)) \qquad \text{se } c \text{ for constante} && (10)\\
O(O(f(n))) &= O(f(n)) && (11)\\
O(f(n)) \cdot O(g(n)) &= O(f(n) \cdot g(n)) && (12)\\
O(f(n) \cdot g(n)) &= f(n) \cdot O(g(n)) && (13)
\end{aligned}
$$

Transitividade:

$$
\begin{aligned}
f(n) = \Theta(g(n)),\quad g(n) = \Theta(h(n)) &\Rightarrow f(n) = \Theta(h(n)) && (14)\\
f(n) = O(g(n)),\quad g(n) = O(h(n)) &\Rightarrow f(n) = O(h(n)) && (15)\\
f(n) = \Omega(g(n)),\quad g(n) = \Omega(h(n)) &\Rightarrow f(n) = \Omega(h(n)) && (16)
\end{aligned}
$$

Reflexividade

$$
\begin{aligned}
f(n) &= \Theta(f(n)) && (17)\\
f(n) &= O(f(n)) && (18)\\
f(n) &= \Omega(f(n)) && (19)
\end{aligned}
$$

Simetria

$$f(n) = \Theta(g(n)) \Leftrightarrow g(n) = \Theta(f(n)) \tag{20}$$

Simetria transposta

$$f(n) = O(g(n)) \Leftrightarrow g(n) = \Omega(f(n)) \tag{21}$$

Pelas propriedades que valem para a notação assintótica, podemos fazer uma analogia entre a comparação assintótica de duas funções f e g e a comparação de dois números reais a e b:

$$
\begin{aligned}
f(n) = O(g(n)) &\approx a \leq b\\
f(n) = \Theta(g(n)) &\approx a = b\\
f(n) = \Omega(g(n)) &\approx a \geq b
\end{aligned}
$$

Há um detalhe: enquanto podemos comparar quaisquer números a e b com as relações $\leq$, $=$ e $\geq$, nem todas as funções podem ser usadas para comparações com O, Θ e Ω. Por exemplo, não podemos escrever algo como $\sin(x) = O(\cos(x))$.

Exemplos

Primeiro teste

Suponha a função

$$f(n) = \frac{n^2}{2} - 3n.$$

Desejamos verificar se a expressão $f(n) = \Theta(n^2)$ é verdadeira. Para tanto, devemos achar constantes c_1, c_2 e n_0 tais que

$$c_1 \cdot n^2 \leq \frac{1}{2}n^2 - 3n \leq c_2 \cdot n^2.$$

Podemos começar dividindo todas as expressões por n^2, obtendo-se

$$c_1 \leq \frac{1}{2} - \frac{3}{n} \leq c_2.$$

Separando as duas desigualdades, resolvemos a primeira e obtemos

$$c_1 \leq \frac{1}{2} - \frac{3}{n}.$$

Pode-se ver que a expressão do lado direito cresce à medida que n cresce, portanto ela assume seu menor valor com $n = 1$ (só nos interessamos para valores positivos de n). Para $n = 1$ teríamos $c_1 \leq -5/2$, mas uma vez que pela definição c_1 deve ser positivo, os valores $n = 1 \ldots 6$ não servem. Finalmente, com $n = 7$ podemos resolver obtendo $c_1 \leq 1/14$, já positivo. Para a segunda desigualdade

$$\frac{1}{2} - \frac{3}{n} \leq c_2,$$

verificamos facilmente que qualquer valor c_2 maior do que $1/2$ satisfaz a desigualdade. Achados c_1, c_2 e n_0 (valendo $1/14$, $1/2$ e 7, respectivamente), o problema está resolvido e a afirmação é verdadeira. É interessante notar que outras constantes poderiam ser usadas, e mesmo outro valor de n_0, mas o importante é que existe alguma combinação satisfatória.

Segundo teste

Podemos usar a definição para testar a afirmação de que $12n^3 \neq O(n^2)$. Para testar, suponhamos que existam c_2 e n_0 tais que $12n^3 \leq c_2 \cdot n^2$ para todo $n \geq n_0$. Dividindo por n, teríamos que $12n \leq c_2$, o que é absurdo já que c_2 será constante e algum valor de n terá de ser maior.

De forma geral, pode-se perceber que os termos de menor ordem de uma função assintoticamente positiva podem ser ignorados na notação assintótica, uma vez que eles passam a ser insignificantes para n grande e uma pequena fração do termo mais significativo é suficiente para dominar os termos de menor ordem. Também podemos ignorar o coeficiente do termo de mais alta ordem, uma vez que ele apenas altera o valor das constantes c_1 e c_2 por um fator igual ao coeficiente.

Como exemplo, tome a função $f(n) = an^2 + bn + c$, onde a, b e c são constantes e $a > 0$. Descartando os termos de menor ordem e ignorando a constante a, temos $f(n) = \Theta(n^2)$.

No caso especial de uma função constante, (por exemplo $f(n) = 17$), podemos expressá-la como $\Theta(n^0)$ ou ainda $\Theta(1)$. Esta última notação, no entanto, esconde qual a variável está indo para infinito, por isso é menos aconselhável.

Revendo (1) e (2)

Usando as regras (13) e (10), podemos verificar que as expressões (1) e (2), no começo de nosso texto, podem ser escritas de outra forma. Acompanhe a dedução para a expressão (1) (analisando apenas o termo que contém a notação O):

$$O\left(\frac{1}{120n^4}\right) = \frac{1}{120}O\left(\frac{1}{n^4}\right) = O\left(\frac{1}{n^4}\right).$$

Assim, as expressões (1) e (2) poderiam ter seus últimos termos simplificados para $O(1/n^4)$ e $O(1/n)$, respectivamente.

Exercícios

1. Para os conjuntos abaixo, verifique se as funções deveriam mesmo pertencer aos conjuntos:

   (a) $O(n^2) = \left\{n, n^{2.5}, \sqrt{n}, \frac{n^3+1}{n}, n^2+2, n^3+n, \ldots\right\}$

   (b) $O(n) = \left\{n, n^{2.5}, \sqrt{n}, \frac{n+1}{n}, n^2+2, n^3+n, \ldots\right\}$

2. Mostre que para quaisquer constantes reais a e b, com $b > 0$, vale

$$(n+a)^b = O(n^b).$$

3. Você sabe que $a = O(f(n))$ e que $b = O(f(n))$. (Qual a forma exata de $f(n)$ não é importante para este problema). Se você tem estes fatos, verifique quais das afirmações abaixo são verdadeiras:

   (a) $a * b = O(f(n^2))$
   (b) $a + b = O(f(n))$
   (c) $a - b = O(0)$
   (d) $1/(a+b) = O(1/f(n))$
   (e) $a * b = O(f(n)^2)$
   (f) $a/b = O(1)$
   (g) $a - b = O(f(n))$
   (h) $1/(a+b) = O(f(1/n))$

4. Neste exercício, qualquer função pode ser usada para fazer o papel de $f(n)$, $g(n)$ ou $h(n)$, por isso os resultados que você obtiver deve ser genéricos, valendo para quaisquer funções. Prove ou negue as seguintes afirmações:

   (a) $f(n) + g(n) = \Theta(\min(f(n), g(n)))$
   (b) $f(n) = O(f(n^2))$
   (c) $f(n) = O(f(2n))$
   (d) $f(n) + g(n) + h(n) = O(\max(f(n), g(n), h(n)))$

(e) $f(n) = O(2f(n))$

5. Analise as consequ"ências da propriedade (17).

6. Analise se $\sin(x) = O(\cos(x))$.

7. Escreva algoritmos que resolvem os problemas abaixo, apresentando estimativas para seu tempo de execução[2]:

   (a) Escreva um subprograma `copy(A,B)` que faz cópias de árvores binárias (de `A` para `B`, por exemplo).

   (b) Escreva um subprograma que recebe um valor real x, um vetor de coeficientes reais $a_0 \ldots a_n$ e avalia o polinômio $P(x) = a_n * x^n + a_{n-1} * x^{n-1} + \cdots + a_1 * x + a_0$.

   (c) Escreva o subprograma que recebe um inteiro n e calcula $n!$

   (d) Escreva o subprograma que recebe um inteiro n escrito em base decimal e o converte para binário.

   (e) Escreva um subprograma que recebe um valor real x, um inteiro n e calcula x^n.

   (f) Escreva um subprograma que recebe um inteiro p e determina se ele é primo.

   (g) Escreva um subprograma que recebe n e avalia $\sum_{k=1}^{n} k^2$.

   (h) Escreva o subprograma que faz a fusão de duas listas encadeadas já ordenadas.

   (i) Escreva um subprograma `mult(A,B)` que recebe duas matrizes `A` e `B`, as multiplica e devolve a matriz resultante.

[2]Lembre que dois algoritmos que resolvem o mesmo problema podem ter estimativas de desempenho diferentes (portanto cuidado ao comparar respostas com colegas!), e lembre **sempre** de dizer qual a variável sendo analisada no problema (Ou seja, explique quem é n no problema estudado!). Algumas vezes, várias variáveis podem estar envolvidas!