Crecimiento de funciones

Elaboró: Dr. Luis Gerardo de la Fraga
Cinvestav, Departamento de Computación

16 de septiembre de 2021

Resumen

Este es una traducción propia del capítulo 3 del libro *Introduction to Algorithms* de T.H. Comen, C.E. Leiserson, R.L. Rivest, C. Stein. 3rd Edition. MIT Press.

1. La notación asintótica

El orden de crecimiento del tiempo de ejecución de un algoritmo nos da una caracterización de la eficiencia del algoritmo y también nos da ayuda a comparar el rendimiento relativo de algoritmo alternativos.

Las funciones que hemos usado, su dominio son el conjunto de números naturales $\mathbb{N} = \{0, 1, 2, \dots\}$. Estas notaciones son útiles para representar al función del tiempo de ejecución en el peor caso, $T(n)$, la cual generalmente se define con tamaños de entrada en enteros.

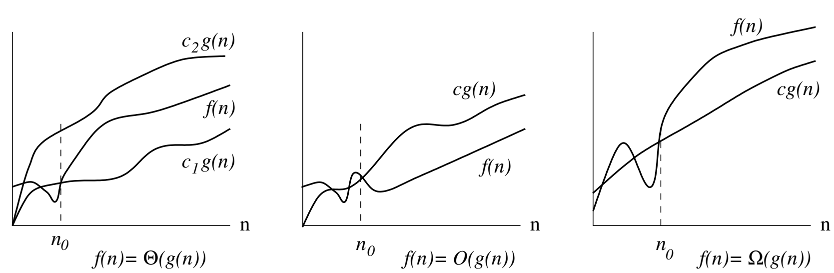
Con el algoritmo de ordenamiento por inserción su peor tiempo de ejecución es $\Theta(n^2)$

Para el ordenamiento por mezcla, su peor tiempo de ejecución es $\Theta(n \lg n)$

Necesitamos comprender que significa esto y como manejar estas expresiones si aparecen en expresiones matemáticas.

1.1. Notación Θ

Para una función dada $g(n)$, se denota por $\Theta(g(n))$ al *conjunto de funciones*



$$\Theta(g(n)) = \{f(n) : \text{ existen constantes positivas } c_1, c_2 \text{ y } n_0 \text{ tal que}$$
$$0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n) \text{ para todo } n \geq n_0\}$$

Una función $f(n)$ pertenece al conjunto $\Theta(g(n))$ si existen constantes positivas c_1 y c_2 tal que forman un sándwich entre $c_1 g(n)$ y $c_2 g(n)$, para un n lo suficientemente grande. Debido a que $\Theta(g(n))$ es un conjunto, se debería escribir "$f(x) \in \Theta(g(n))$"para indicar que $f(x)$ es un elemento de $\Theta(g(n))$. En vez de esto, usualmente se escribe: "$f(x) = \Theta(g(n))$"para expresar la misma noción.

De la figura 1.1, se dice que $g(x)$ es una **cota ajustada asintóticamente** para $f(x)$.

La definición para $\Theta(n)$ requiere que cada función miembro $f(n) \in \Theta(g(n))$ sea **asintóticamente no negativa**, esto es, que $f(n)$ sea no negativa para cualquier n lo suficientemente grande. Consecuentemente la función $g(n)$ es sí misma tiene que ser no negativa asintóticamente, o el conjunto $\Theta(g(n))$ está vacío. Entonces asumiremos que todas las funciones usadas en la notación $\Theta(g(n))$ son no negativas asintóticamente.

Vamos a ver un ejemplo. Tenemos que mostrar que:

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

Para esto, tenemos que buscar dos constantes c_1 y c_2 tal que

$$c_1 n^2 \leq \frac{n^2}{2} - 3n \leq c_2 n^2,$$

para todo $n \geq n_0$. Dividimos entre n^2 :

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

Para la inecuación de la derecha:

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

tenemos

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

n	$f(n)$
1	-2/2 = -1
2	-1/4
3	0
4	1/8
∞	1/2

para cualquier valor $n \geq 1$, $c_2 \geq 1/2$.

Para el lado izquierdo:

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

n	$f(n)$
5	$\frac{-1}{10}$
6	$\frac{0}{12} = 0$
7	$\frac{1}{14}$
8	$\frac{1}{8}$
9	$\frac{1}{6}$

y entonces $c_1 \leq 1/14$ para $n_0 >= 7$

Y entonces verificamos que

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

Ciertamente se pueden escoger otros valores para c_1 y c_2, pero lo importante aquí es demostrar que esas constantes existen. Estas constantes dependen de la función $n^2/2 - 3n$; para otras funciones que pertenezcan a $\Theta(n^2)$ podrían requerir valores de las constantes distintas.

Otro ejemplo. Verificamos que $6n^3 \neq \Theta(n^2)$. Debemos de mostrar que:

$$c_1 n^2 \leq 6n^3 \leq c_2 n^2, \text{ para todo } n \geq n_0$$

Dividimos entre n^2

$$c_1 \leq 6n \leq c_2, \text{ para todo } n \geq n_0$$

Para el lado derecho tenemos:

$$6n \leq c2$$
$$n \leq \frac{c_2}{6}$$

y esto no se puede cumplir para n grandes, dado que c_2 es una constante.

Para cualquier polinomio

$$p(n) = \sum_{i=0}^{d} a_i n^i$$

se tiene

$$p(n) = \Theta(n^d)$$

Si $d = 3$:

$$p(n) = a_3 n^3 + a_2 n^2 + a_1 n + a_0$$

y

$$p(n) = \Theta(n^3)$$

Nos faltan ver los notaciones O, Ω, o y ω.

1.2. Notación O

Cuando solo se tiene la cota superior asintótica, se usa la notación O. Para una función $g(n)$, se escribe $O(g(n))$ ("La O grande de g de n") al conjunto de funciones

$$O(g(n)) = \{f(n) : \text{ existen constantes positivas } c \text{ y } n_0 \text{ tal que } \\ 0 \leq f(n) \leq cg(n) \text{ para todo } n \geq n_0\}$$

Se usan la notación O para dar una cota superior a una función, dentro de una factor constante

Se escribe $f(n) = O(g(n))$ para indicar que una función $f(n)$ es miembro del conjunto $O(g(n))$.

Nótese que $f(n) = \Theta(g(n))$ implica $f(n) = O(g(n))$, dado que la notación Θ es más poderosa que la O. Matemáticamente $\Theta(g(n)) \subseteq O(g(n))$.

Entonces una prueba de que cualquier función cuadrática $an^2 + bn + c$ para $a > 0$, está en $\Theta(n^2)$ también muestra que está función está en $O(n^2)$.

Al usar la notación O se puede describir el tiempo de ejecución de un algoritmo solamente por inspección de la estructura del algoritmo. Por ejemplo, el ciclo doblemente anidado del algoritmo de ordenamiento por inserción que ya vimos, inmediatamente resulta en una cota superior $O(n^2)$ sobre el tiempo de ejecución en el peor caso: el costo de cada iteración en el ciclo interior está acotado por el exterior por $O(1)$ (es constante), los índices i y j son ambos a los más n y el ciclo interior es ejecutado a lo más una vez para uno de los n^2 pares de valores de i y j.

Ya que la notación O describe una cota superior, cuando se usa esta para acotar el tiempo de ejecución del peor caso de un algoritmo, se tiene una cota del tiempo de ejecución del algoritmo para cada entrada. Entonces, la cota $O(n^2)$ en peor caso de ejecución del algoritmo de ordenamiento por inserción también implica al tiempo de ejecución sobre cada entrada. La cota $\Theta(n^2)$ en el peor tiempo de ejecución para el algoritmo de inserción, sin embargo, no implica a la cota $\Theta(n^2)$ sobre el tiempo de ejecución del algoritmo de ordenamiento por inserción para *cada* entrada. Por ejemplo, ya vimos que para una entrada ya ordenada, su tiempo de ejecución es en tiempo $\Theta(n)$.

Técnicamente, es un abuso decir que el tiempo de ejecución del ordenamiento por inserción es $O(n^2)$, ya que para cada n, el tiempo de ejecución varía, dependiendo sobre el tamaño particular de la entrada de tamaño n. Cuando se dice "el tiempo de ejecución es $O(n^2)$", se quiere decir que hay una función $f(n)$ que es $O(n^2)$ tal que para cualquier valor de n, sin importar de cuál tamaño particular de n se escoja, el tiempo de ejecución sobre esa entrada está acotado por arriba por el valor de $f(n)$. Equivalentemente, significa también que el el tiempo de ejecución en el peor caso es $O(n^2)$.

1.3. Notación Ω

Tal como la notación O provee una cota superior asintótica *superior*, la notación Ω provee una **cota asintótica inferior**. Para una función dada $g(n)$, se denota $\Omega(g(n))$ (se pronuncia como “omega grande de g de n”, o algunas veces solo “omega de g de n”) al conjunto de funciones

$$\begin{aligned}\Omega(g(n)) = \{f(n) :\ & \text{existen constantes positivas } c \text{ y } n_0 \text{ tal que} \\ & 0 \le cg(n) \le f(n) \text{ para todo } n \ge n_0\}\end{aligned}$$

La figura 1.1 muestra la intuición atrás de la notación Ω. Para todos los valores de n en o a la derecha de n_0, el valor de $f(n)$ está en o arriba de $cg(n)$.

Teorema 1. *Para cualquiera dos funciones $f(n)$ y $g(n)$, se tiene $f(n) = \Theta(g(n))$ si y solo si $f(n) = O(g(n))$ y $f(n) = \Omega(g(n))$.* ■

Como un ejemplo de la aplicación de este teorema, la prueba de que $an^2 + bn + c = \Theta(n^2)$ para las constantes cualquiera a, b y c, donde $a > 0$, implica inmediatamente que $an^2+bn+c = \Omega(n^2)$ y $an^2+bn+c = O(n^2)$. En la práctica mas que usar el teorema 1 para obtener las cotas superior e inferior para cotas ajustadas asintóticamente, como se hizo en el ejemplo, se usan para probar cotas ajustadas asintóticamente a partir de las cotas asintóticas superior e inferior.

Cuando de dice que el *tiempo de ejecución* de un algoritmo es $\Omega(g(n))$, significa que *sin importar que entrada particular de tamaño n se escoge para cada valor de n*, el tiempo de ejecución para cada entrada es al menos una constante de tiempo $g(n)$, para n lo suficientemente grandes. Equivalentemente, se está dando una cota inferior sobre el tiempo de ejecución en el mejor caso para un algoritmo. Por ejemplo, el tiempo de ejecución en el mejor caso para el ordenamiento por inserción es $\Omega(n)$.

El tiempo de ejecución para el ordenamiento por inserción por lo tanto pertenece tanto a $\Omega(n)$ como a $O(n^2)$, ya que yace en cualquier parte entre una función lineal de n y una función cuadrática de n. Aún más, estas cotas son asintóticamente tan ajustadas como es posible: por ejemplo, el tiempo de ejecución para el ordenamiento por inserción no es $\Omega(n^2)$, ya que existe una entrada para la cual el ordenamiento por inserción corre en tiempo $\Omega(n)$ (esto es, cuando la entrada ya está ordenada). No es contradictorio decir, sin embargo, que el tiempo de ejecución *en el peor caso* para el ordenamiento por inserción es $\Omega(n^2)$, ya que existe una entrada que causa que el algoritmo tome tiempo $\Omega(n^2)$.

1.4. Notación asintótica en ecuaciones e inecuaciones

Ya hemos visto que la notación asintótica puede aparecer en fórmulas matemáticas. Por ejemplo, en la introducción a la notación O, se escribió “$n = O(n^2)$”. Se podría escribir también $2n^2 + 3n + 1 = 2n^2 + \Theta(n)$. ¿Cómo se deberían interpretar estas fórmulas?

Cuando aparece la notación asintótica aislada (esto es, sin estar dentro de una fórmula más grande) sobre el lado derecho de un ecuación (o de una inecuación), tal como en $n = O(n^2)$, ya se ha definido que el signo de igual aquí significa pertenencia a un conjunto: $n \in O(n^2)$. En general, sin embargo, cuando la notación simbólica aparece en una fórmula, se interpreta esta como el inicio para alguno fórmula anónima en la cual no nos importa el nombre. Por ejemplo, la fórmula

$$2n^2 + 3n + 1 = 2n^2 + \Theta(n)$$

significa que

$$2n^2 + 3n + 1 = 2n^2 + f(n),$$

para $f(n)$ es alguna función en $\Theta(n)$. En este caso, se hace $f(n) = 3n + 1$, la cuál de hecho está en $\Theta(n)$

Al usar la notación asintótica de esta manera se pueden eliminar detalles inesenciales y confusiones en una ecuación. Por ejemplo, cuando se expresó el tiempo de ejecución en el peor caso del ordenamiento por mezcla como la recurrencia

$$T(n) = 2T(n/2) + \Theta(n).$$

Si solamente se está interesado en el comportamiento asintótico de $T(n)$, no hay punto en especificar todos los términos orden bajo exactamente; se sobreentienden que se incluyen en la función anónima denotada por el término $\Theta(n)$.

El número de funciones anónimas en una expresión se sobreentiende que es igual al número de veces que la función asintótica aparece. Por ejemplo, en la expresión

$$\sum_{i=1}^{n} O(i),$$

hay solamente una función anónima (una función de i). Esta expresión no es lo mismo que $O(1) + O(2) + \cdots + O(n)$, la cual no tiene una interpretación clara.

En algunos casos, la notación asintótica aparece en el lado izquierdo de una ecuación como en

$$2n^2 + \Theta(n) = \Theta(n^2).$$

Se interpreta esta ecuación usando la siguiente regla: *sin importar como se escojan las funciones anónimas sobre el lado izquierdo del signo de igual, hay una manera de escoger la función anónima sobre el lado derecho del signo de igual para hacer la ecuación válida.* Por lo que, para nuestro ejemplo significa que cualquier función $f(n) \in \Theta(n)$, hay una función $g(n) \in \Theta(n^2)$ tal que $2n^2 + f(n) = g(n)$ para todo n. En otras palabras, el lado derecho de la función de una ecuación provee un nivel más grueso de detalle que el lado izquierdo.

Se pueden encadenar juntos un número de tales relaciones, como en

$$\begin{aligned} 2n^2 + 3n + 1 &= 2n^2 + \Theta(n) \\ &= \Theta(n^2). \end{aligned}$$

Se puede interpretar cada ecuación separadamente por los reglas de arriba. La primera ecuación dice que hay alguna función $f(n) \in \Theta(n)$ tal que $2n^2+3n+1 = 2n^2+f(n)$ para todo n. La segunda ecuación dice que para cualquier función $g(n) \in \Theta(n)$ (junto como la función ya mencionada), hay alguna función $h(n) \in \Theta(n^2)$ tal que $2n^2 + g(n) = h(n)$ para todo n. Nótese que esta interpretación implica que $2n^2 + 3n + 1 = \Theta(n^2)$, lo cual es la encadenación de la ecuación intuitivamente nos da.

1.5. Notación o

La cota superior asintótica que provee la notación O puede o no ser asintóticamente ajustada. La cota $2n^2 = O(n^2)$ es asintóticamente ajustada, pero la cota $2n = O(n^2)$ no lo es. Se usa la notación o para denotar una cota superior que no es asintóticamente ajustada. Se define formalmente $o(g(n))$ como el conjunto

$$o(g(n)) = \{f(n) : \text{ para cualquier constante positiva } c > 0, \text{ existe una constante } n_0 > 0 \text{ tal que } 0 \leq f(n) < cg(n) \text{ para todo } n \geq n_0\}$$

Por ejemplo: $2n = o(n^2)$, pero $2n^2 \neq o(n^2)$

Las definiciones para la notación O y la notación o son similares. La diferencia principal es que en $f(n) = O(g(n))$, la cota $0 \leq f(n) \leq cg(n)$ se mantiene para *alguna* constante $c > 0$, pero en $f(n) = o(g(n))$, la cota $0 \leq f(n) < cg(n)$ se mantiene para *todas* las constantes $c > 0$. Intuitivamente, en la notación o, la función $f(n)$ se vuelve insignificante relativa a $g(n)$ cuando n se aproxima al infinito; esto es

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

1.6. Notación ω

Por analogía, la notación ω es a la notación Ω como la notación o es a la notación O. Se usa esta notación para denotar una cota inferior que no es asintóticamente ajustada. Una forma de definirla es

$$f(n) \in \omega(g(n)) \text{ si y solo si } g(n) \in o(f(n))$$

Formalmente, sin embargo, se define como el conjunto

$$\omega(g(n)) = \{f(n) : \text{ para cualquier constante positiva } c > 0, \text{ existe una constante } n_0 > 0 \text{ tal que } 0 \leq cg(n) < f(n) \text{ para todo } n \geq n_0\}$$

Por ejemplo, $n^2/2 = \omega(n)$, pero $n^2/2 \neq \omega(n^2)$. La relación $f(n) = \omega(g(n))$ implica que

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \infty ,$$

si el límite existe. Esto es, $f(n)$ se vuelve arbitrariamente grande relativa a $g(n)$ cuando n se aproxima al infinito.

1.7. Comparación de funciones

Muchas de las relaciones que se aplican a los números reales también se aplican a las comparaciones asintóticas. En lo sigue se asume que $f(n)$ y $g(n)$ son asintóticamente positivas

Transitividad:

$$
\begin{aligned}
f(n) = \Theta(g(n)) \text{ y } g(n) = \Theta(h(n)) &\text{ implica } f(n) = \Theta(h(n)).\\
f(n) = O(g(n)) \text{ y } g(n) = O(h(n)) &\text{ implica } f(n) = O(h(n)).\\
f(n) = \Omega(g(n)) \text{ y } g(n) = \Omega(h(n)) &\text{ implica } f(n) = \Omega(h(n)).\\
f(n) = o(g(n)) \text{ y } g(n) = o(h(n)) &\text{ implica } f(n) = o(h(n)).\\
f(n) = \omega(g(n)) \text{ y } g(n) = \omega(h(n)) &\text{ implica } f(n) = \omega(h(n)).
\end{aligned}
$$

Reflexividad:

$$
\begin{aligned}
f(n) &= \Theta(f(n)).\\
f(n) &= O(f(n)).\\
f(n) &= \Omega(f(n)).\\
f(n) &= o(f(n)).\\
f(n) &= \omega(f(n)).
\end{aligned}
$$

Simetría:

$f(n) = \Theta(g(n))$ si y solo si $g(n) = \Theta(f(n))$.

Simetría transpuesta:

$$
\begin{aligned}
f(n) = O(g(n)) &\text{ si y solo si } g(n) = \Omega(f(n))\\
f(n) = o(g(n)) &\text{ si y solo si } g(n) = \omega(f(n))
\end{aligned}
$$

Debido a que se mantienen estas propiedades para la notación asintótica, se puede escribir por analogía la comparación asintótica entre dos funciones f y g las comparaciones de dos números reales a y b:

$$
\begin{aligned}
f(n) = O(g(n)) &\text{ es como } a \leq b,\\
f(n) = \Omega(g(n)) &\text{ es como } a \geq b,\\
f(n) = \Theta(g(n)) &\text{ es como } a = b,\\
f(n) = o(g(n)) &\text{ es como } a < b,\\
f(n) = \omega(g(n)) &\text{ es como } a > b.
\end{aligned}
$$

Entonces de dice que $f(n)$ es **asintóticamente menor** que g(n) si $f(n) = o(g(n))$, y $f(n)$ es **asintóticamente mayor** que $g(n)$ si $f(n) = \omega(g(n))$.

Hay una propiedad de los números reales que no se mantiene sobre la notación asintótica:

Tricotomía: Para cualquiera dos números reales a y b, exactamente una de las siguientes relaciones se debe mantener: $a < b$, $a = b$ o $a > b$.

Aunque dos números reales se pueden comparar, no todas las funciones asintóticas son comparables. Esto es, para dos funciones $f(n)$ y $g(n)$, puede ser el caso de que ni $f(n) = O(g(n))$ ni $f(n) = \Omega(g(n))$ se mantengan. Por ejemplo, no podemos comparar las funciones n y $n^{1+\operatorname{sen} n}$ usando la notación asintótica, ya que que el valor del exponente en $n^{1+\operatorname{sen} n}$ varía entre 0 y 2 y todos su valores están dentro de este intervalo.

2. Notación estándar y funciones comunes

Es esta sección se revisará algunas funciones matemáticas estándar y notaciones y explora la relación entre ellas. También se ilustra el uso de la notación asintótica.

Monoticidad

Una función $f(n)$ es **monotónicamente creciente** si $m \leq n$ implica $f(m) \leq f(n)$. De forma similar esta es **monotónicamente decreciente** if $m \leq n$ implica $f(m) \geq f(n)$. Una función $f(n)$ es **estrictamente creciente** si $m < n$ implica $f(m) < f(n)$ y es **estrictamente decreciente** si $m < n$ implica $f(m) > f(n)$.

2.1. Piso y techo

Para cualquier número real x, se denota al entero más grande menor o igual a x por $\lfloor x \rfloor$ (que se lee como "el piso de x") y el entero menor más grande que o igual x por $\lceil x \rceil$ (que se lee como "el techo de x"). Para todo real x,

$$x - 1 < \lfloor x \rfloor \leq x \leq \lceil x \rceil < x + 1.$$

Para cualquier entero n,

$$\lceil n/2 \rceil + \lfloor n/2 \rfloor = n,$$

y para cualquier número real $x \geq 0$ y enteros $a, b > 0$,

$$\begin{aligned}
\left\lceil \frac{\lceil x/a \rceil}{b} \right\rceil &= \left\lceil \frac{x}{ab} \right\rceil, \\
\left\lfloor \frac{\lfloor x/a \rfloor}{b} \right\rfloor &= \left\lfloor \frac{x}{ab} \right\rfloor, \\
\left\lceil \frac{a}{b} \right\rceil &\leq \frac{a + (b-1)}{b}, \\
\left\lfloor \frac{a}{b} \right\rfloor &\geq \frac{a - (b-1)}{b}.
\end{aligned}$$

La función piso $f(x) = \lfloor x \rfloor$ es monotónicamente creciente, como también lo es la función techo $f(x) = \lceil x \rceil$.

2.2. Aritmética modular

Para cualquier entero a y cualquier entero positivo n, el valor $a \bmod n$ es el **residuo** de la razón a/n:

$$a \bmod n = a - n\lfloor a/n \rfloor.$$

De aquí sigue que

$$0 \leq a \bmod n < n.$$

Dada una noción bien definida del residuo de un entero cuando es dividido por otro, es conveniente proveer una notación especial para indicar la igualdad de residuos. Si $(a \bmod n) = (b \bmod n)$, se escribe $a \equiv b(\bmod n)$ y se dice que a es **equivalente** a b, módulo n. En otras palabras, $a \equiv b(\bmod n)$ tienen el mismo residuo cuando son divididos por n. Equivalentemente, $a \equiv b(\bmod n)$ si y solo si n es un divisor de $b - a$. Se escribe $a \not\equiv b(\bmod n)$ si a no es equivalente a b, módulo n.

Polinomios

Dado un entero no negativo d, un **polinomio en** n **de grado** d es una función $p(n)$ de la forma

$$p(n) = \sum_{i=0}^{d} a_i n^i,$$

donde las constantes $a_0, a_1, \ldots, a_n$ son los **coeficientes** del polinomio y $a_d \neq 0$. Un polinomio es asintóticamente positivo si y solo si $a_d > 0$. Para un polinomio asintóticamente positivo $p(n)$ de grado d, se tiene $p(n) = \Theta(n^d)$. Para cualquier constante $a \geq 0$, la función n^a es monotónicamente creciente, y para cualquier constante real $a \leq 0$, la función n^a es monotónicamente decreciente. Se dice que la función $f(n)$ es **acotada polinómicamente** si $f(n) = O(n^k)$ para alguna constante k.

Exponenciales

Para todo real $a > 0$, m y n, se tienen las siguientes identidades:

$$\begin{aligned}
a^0 &= 1, \\
a^1 &= a, \\
a^{-1} &= 1/a, \\
(a^m)^n &= a^{mn}, \\
(a^m)^n &= (a^n)a^m, \\
a^m a^n &= a^{n+m}.
\end{aligned}$$

Para todo n y $a \geq 1$, la función a^n es monotónicamente creciente en n. Cuando sea conveniente, se asumirá que $0^0 = 1$.

Se puede relacionar la razón de crecimiento de polinomios y exponenciales con el siguiente hecho. Para todas las constantes reales a y b tal que $a > 1$,

$$\lim_{n\to\infty} \frac{n^b}{a^n} = 0, \tag{1}$$

de lo cual se concluye que

$$n^b = o(a^n).$$

Entonces, cualquier función exponencial con un base estrictamente más grande que 1 crece más rápido que cualquier función polinomial.

Usando e para denotar $2.71828\ldots$ como la base de la función logaritmo natural, se tiene que para cualquier real x,

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots = \sum_{i=0}^{\infty} \frac{x^i}{i!},$$

donde "!"denota la función factorial que se definirá más tarde. Para todo real x se tiene la inegualdad

$$e^x \geq 1 + x,$$

donde la igualdad se mantiene solo cuando $x = 0$. Cuando $|x| \leq 1$, se tiene la aproximación

$$1 + x \leq e^x \geq 1 + x + x^2.$$

Cuando $x \to 0$, la aproximación de e^x por $1 + x$ es muy buena:

$$e^x = 1 + x + \Theta(x^2).$$

(En esta ecuación, la notación asintótica se usa para describir el comportamiento límite cuando $x \to 0$ mas que cuando $x \to \infty$). Para todo x se tiene,

$$\lim_{n\to\infty} \left(1 + \frac{x}{n}\right)^n = e^x.$$

Logaritmos

Se usará las siguientes notaciones:

$$\begin{aligned}
\lg n &= \log_2 n \quad \text{(logaritmo binario)},\\
\ln n &= \log_e n \quad \text{(logaritmo natural)},\\
\lg^k n &= (\lg n)^k \quad \text{(exponenciación)},\\
\lg\lg n &= \lg(\lg n) \quad \text{(concatenación)}.
\end{aligned}$$

Una convención notacional importante que se adoptará es que las *funciones logaritmo se aplicarán solo al siguiente término de la fórmula*, de forma que $\lg n + k$ significará $(\lg n) + k$ y no $\lg(n + k)$. Si se mantiene $b > 1$ constante, entonces para $n > 0$, las función $\log_b n$ es estrictamente creciente.

Para todo real $a > 0$, $b > 0$, $c > 0$ y n,

$$\begin{aligned} a &= b^{\log_b a}, \\ \log_c(ab) &= \log_c a + \log_c b, \\ \log_c a^n &= n \log_c a, \\ \log_b a &= \frac{\log_c a}{\log_c b}, & (2) \\ \log_b(1/a) &= -\log_b a, \\ \log_b a &= \frac{1}{\log_a b}, \\ a^{\log_b c} &= c^{\log_b a}, & (3) \end{aligned}$$

donde, en cada una de las ecuaciones de arriba, las bases de los logaritmos no son 1.

De la ecuación (2), el cambiar la base de los logaritmos de una constante a otra cambia el valor del logaritmo solo por un factor constante, de forma que se usará la notación "$\lg n$" cuando no nos preocupamos sobre los factores constantes, tal como en la notación O. Los científicos en computación encuentran 2 como la base más natural para los logaritmos porque muchos algoritmos y estructuras de datos envuelven dividir un problema en dos partes.

Hay una expansión en series simple para $\ln(1+x)$ cuando $|x| < 1$:

$$\ln(1+x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \frac{x^5}{5} - \ldots.$$

Se tiene también la siguiente inegualdad para $x > -1$:

$$\frac{x}{1+x} \leq \ln(1+x) \leq x,$$

donde la igualdad se mantiene solo para $x = 0$.

Se dice que la función $f(n)$ es **polinómicamente acotada** si $f(n) = O(\lg^k n)$ para alguna constante k. Se puede relacionar el crecimiento de polinomios y polilogaritmos sustituyendo $\lg n$ por n y 2^a por a en la ec. (1), resultando

$$\lim_{n\to\infty} \frac{\lg^b n}{(2^a)^{\lg n}} = \lim_{n\to\infty} \frac{\lg^b n}{n^a} = 0.$$

Por este límite se puede concluir que

$$\lg^b n = o(n^a)$$

para cualquier constante $a > 0$. Entonces, cualquier función polinomial positiva crece más grande que cualquier función polilogarítmica.

Factoriales

La notación $n!$ (leída como "factorial de n") está definida para enteros $n > 0$ como

$$n! = \begin{cases} 1 & \text{si } n = 0, \\ n(n-1)! & \text{si } n > 0; \end{cases}$$

Entonces, $n! = 1 \cdot 2 \cdot 3 \cdots n$.

Una cota superior débil sobre la función factorial es $n! \leq n^n$, ya que cada de los n términos en el producto factorial es a lo más n. La **aproximación de Stirling**,

$$n! = \sqrt{2\pi n}\left(\frac{n}{e}\right)^n \left(1 + \Theta\left(\frac{1}{n}\right)\right),$$

donde e es la base de los logaritmos naturales, da una cota superior más ajustada, y también una cota inferior.

$$\begin{aligned} n! &= o(n^n) \\ n! &= w(2^n) \\ \lg(n!) &= \Theta(n \lg n), \end{aligned}$$

donde la aproximación de Stirling es de ayuda para demostrar las ecuaciones anteriores. La siguiente ecuación también se mantiene para todo $n \geq 1$:

$$n! = \sqrt{2\pi n}\left(\frac{n}{e}\right)^n e^{\alpha_n}$$

donde

$$\frac{1}{12n+1} < \alpha_n \frac{1}{12n}.$$

Iteración funcional

Se usa la notación $f^{(i)}(n)$ para denotar la función $f(n)$ aplicada iterativamente i veces a un valor inicial de n. Formalmente, sea $f(n)$ una función sobre los reales. Para enteros i no negativos, recursivamente se define

$$f^{(i)}(n) = \begin{cases} n & \text{si } i = 0, \\ f(f^{(i-1)}(n)) & \text{si } i > 0; \end{cases}$$

Por ejemplo, si $f(n) = 2n$, entonces $f^{(i)}(n) = 2^i n$

La función logaritmo iterativa

Se usa la notación $\lg^* n$ (que se lee como "log estrella de n") para denotar el logaritmo iterativo. definido como sigue. Sea $\lg^{(i)} n$ definida como arriba, con $f(n) = \lg n$ Porque el logaritmo de un número no positivo está indefinido, $\lg^{(i)} n$ está definido solo si $\lg^{(i-1)} n > 0$. Esté seguro de distinguir $\lg^{(i)} n$ (la función logaritmo aplicada i veces en sucesión, comenzando con el argumento n) de $\lg^i n$

(el logaritmo de n a la i-ésima potencia). Entonces se define la función logaritmo iterativa como

$$\lg^* n = \text{mín}\left\{i \geq 0 : \lg^{(i)} n \leq 1\right\}.$$

El logaritmo iterativo es una función de crecimiento *muy* lento:

$$\begin{aligned}
\lg^* 2 &= 1, \\
\lg^* 4 &= 2, \\
\lg^* 16 &= 3, \\
\lg^* 65536 &= 4, \\
\lg^*(2^{65536}) &= 5.
\end{aligned}$$

Dado que el número de átomos del universo observable está estimado que es cerca de 10^{80}, lo cual es mucho menor que 2^{65536}, raramente se encontrará una entrada de tamaño n que sea $\lg^* n > 5$.

Números de Fibonacci

Se definen los **números de Fibonacci** por la recurrencia siguiente:

$$\begin{aligned}
F_0 &= 0, \\
F_1 &= 1, \\
F_i &= F_{i-1} + F_{i-2} \qquad \text{para } i \geq 2.
\end{aligned}$$

Entonces, cada número de Fibonacci es la suma de los dos previos, resultando en la secuencia

$$0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, \ldots.$$

Estos números están relacionados con la **razón de oro** ϕ y su conjugado $\hat{\phi}$, las cuales son las dos raíces de la ecuación

$$x^2 = x + 1$$

y están dadas por las siguientes fórmulas:

$$\begin{aligned}
\phi &= \frac{1+\sqrt{5}}{2} \\
\phi &= 1.61803\ldots \\
\hat{\phi} &= \frac{1-\sqrt{5}}{2} \\
\phi &= -0.61803\ldots
\end{aligned}$$

Específicamente, se tiene

$$F_i = \frac{\phi^i - \hat{\phi}^i}{\sqrt{5}},$$

que puede probarse por inducción. Ya que $|\hat{\phi}| < 1$, se tiene

$$\begin{aligned} \frac{|\hat{\phi}|}{\sqrt{5}} &< \frac{1}{\sqrt{5}} \\ &< \frac{1}{2}, \end{aligned}$$

lo cual implica que

$$F_i = \left\lfloor \frac{\phi^i}{\sqrt{5}} + \frac{1}{2} \right\rfloor$$

lo cual es decir que el i-ésimo número de Fibonacci F_i es igual a $\phi^i/\sqrt{5}$ redondeado al entero más cercano. Entonces, los números de Fibonacci crecen exponencialmente.