

Recapitulando la demostración vista en clase. Demostramos que el número de vértices en el árbol de ejecución de la MTN N es $O(b^{t(n)})$, y que la simulación con M de N para llegar a cualquier vértice del árbol toma tiempo O(t(n)), por lo que la simulación completa de M toma tiempo $O(t(n)b^{t(n)})$. Si M es de tiempo t'(n), entonces acabamos de observar que t'(n) es $O(t(n)b^{t(n)})$, es decir, $t'(n) \le ct(n)b^{t(n)}$ para alguna constante c y a partir de algún número n_0 . Como la función logaritmo es creciente, tenemos que

$$\log_{b}(t'(n)) \le \log_{b}(ct(n)b^{t(n)})$$

$$= \log_{b}(c) + \log_{b}(t(n)) + \log_{b}(b^{t(n)})$$

$$= \log_{b}(c) + \log_{b}(t(n)) + t(n)$$

Como $\log_b(t(n)) \in O(t(n))$, se sigue que existe una constante c' y un número natural n_1 tales que, si $n \ge n_1$, entonces

$$\log_b(c) + \log_b(t(n)) + t(n) \le c't(n).$$

Por lo tanto, $\log_b(t'(n)) \le c't(n)$, y como la exponenciación es una función creciente, tenemos que

$$t'(n) = b^{\log_b(t'(n))} \le b^{c't(n)},$$

Si $a = \log_2(b)$, entonces $b^{c'} = \left(2^a\right)^{c'} = 2^{ac'}$, por lo que

$$t'(n) \le b^{c't(n)} = (b^{c'})^{t(n)} = (2^{ac'})^{t(n)} = 2^{ac't(n)}$$

Como ac' es una constante, ac't(n) es O(t(n)), por lo que concluimos que $2^{ac't(n)}$ es $2^{O(t(n))}$, y por lo tanto, t'(n) es $2^{O(t(n))}$.

Entonces aclaro:

 No es cierto que cualesquiera dos funciones exponenciales sean equivalentes bajo la notación asintótica. En particular, por ejemplo,

$$O(2^n) \neq O(3^n),$$

pero 2^n sí es $O(3^n)$.

Además, $2^{O(n)}$ no es equivalente a $O(2^n)$. De hecho, es mucho más fuerte $2^{O(n)}$, pues, si $f(n) = 2^{O(n)}$, entonces $\log(f(n)) = O(n)$, es decir, $\log(f(n)) \le cn$, para alguna constante c y una n suficientemente grande, de donde se sigue que $f(n) = 2^{\log(f(n))} \le 2^{cn} = (2^c)^n$. Si $b = 2^c$, entonces concluimos que f(n) es $O(b^n)$. Además, podemos elegir a c tan grande como queramos, por lo que b puede ser tan grande como queramos.