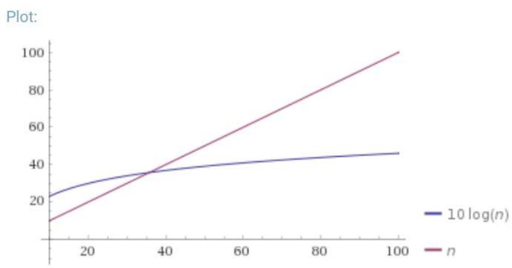
Why is that the definition?

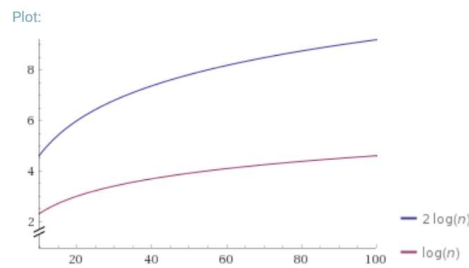
Big-O

$f(n)$ is $O(g(n))$ if there exist positive constants c, n_0 such that for all $n \geq n_0$,
$f(n) \leq c \cdot g(n)$

Why n_0?



Why c?



5

5

O, Omega, Theta [oh my?]

Big-O is an upper bound

- My code uses at most this many resources (e.g. runs in at most this much time)

Big-Omega is a lower bound

Big-Omega

$f(n)$ is $\Omega(g(n))$ if there exist positive constants c, n_0 such that for all $n \geq n_0$,
$f(n) \geq c \cdot g(n)$

Big Theta is "equal to"

Big-Theta

$f(n)$ is $\Theta(g(n))$ if
$f(n)$ is $O(g(n))$ and $f(n)$ is $\Omega(g(n))$.

8

8

Logs

$\log(n^k) = k \cdot \log(n)$.

$\log(2^n) = n$ (logs and exponents are inverse functions.

$\log(\log(x))$ is usually written `log log x`

Grows **VERY** slowly (as slowly as $2^{(2^x)}$ grows quickly)

$\log_2(\log_2(\text{\# atoms in universe})) = \log_2(\log_2(10^{80})) = \log_2(80 \cdot \log_2(10)) \approx 8.054$

Don't confuse with $\log(x) \cdot \log(x)$ usually written $\log^2(x)$.

$\log\log x << \log x << \log^2(x)$ (where << is "asymptotically less than")

Using the Definition

Let's show: $10n^2 + 15n$ is $O(n^2)$