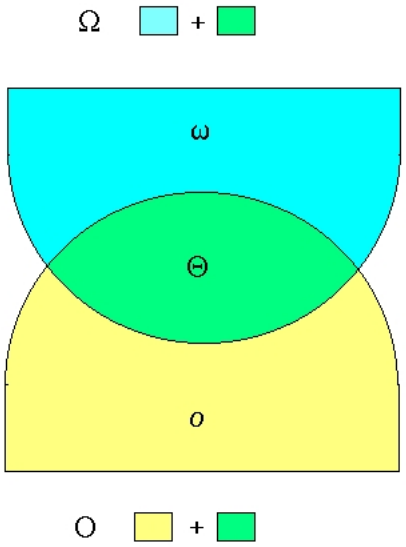
CS201 Data Structures and Algorithms

Order Notation

Rough ideas

Consider this Venn diagram:



We can assign the following meanings:

ω	always worse than	lower bound
o	always better than	upper bound
Ω	never better than	lower bound
O	never worse than	upper bound
Θ	never better/worse than	upper and lower (tight) bound

under the following conditions:

- the problem size (usually denoted n) is sufficiently large (NSL)
- we are comparing worst case behavior (WCB)

For the Θ, Ω, and O regions, we apply one more condition:

- we ignore constant factors and other lower order terms (ICF)

If we are comparing the running times of two algorithms, f and g, we place one of them, say g, in the Θ region. If algorithm f is in the ω region, then:

$$f = \omega(g)$$

The English interpretation is that f is always slower than g (NSL,WCB). If f is in the Θ or o regions, then:

$$f = O(g)$$

The English interpretation of this statement is that f is never slower than g (NSL,WCB,ICF).

When using order notation, one generally assumes that NSL, WCB, and ICF always apply, unless otherwise stated.

Formal Statements

A formal definition of O is:

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

The definitions of the other symbols are similar.