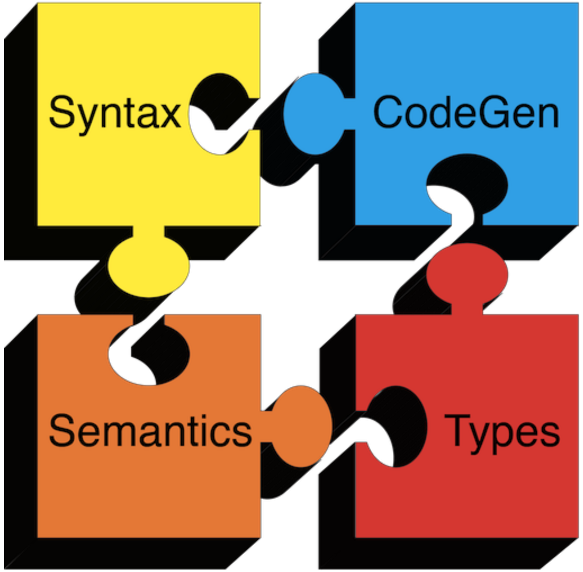


Compilers

Orderings

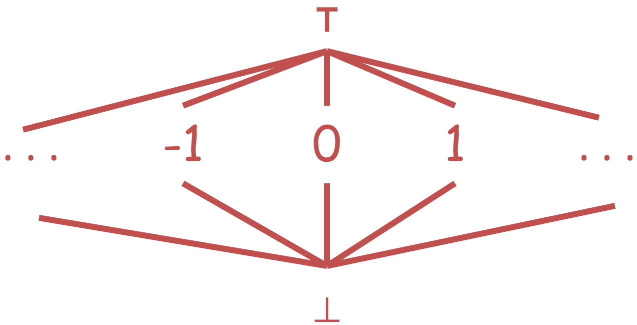
Alex Aiken

Orderings

- We can simplify the presentation of the analysis by ordering the values

⊥ < c < T

- Drawing a picture with “lower” values drawn lower, we get



Orderings

- $\top$ is the greatest value, $\bot$ is the least
  - All constants are in between and incomparable

- Let *lub* be the least-upper bound in this ordering

- Rules 1-4 can be written using lub:

  C(s, x, in) = lub { C(p, x, out) | p is a predecessor of s }

Orderings

- Simply saying “repeat until nothing changes” doesn’t guarantee that eventually nothing changes

- The use of lub explains why the algorithm terminates
  - Values start as $\bot$ and only *increase*
  - $\bot$ can change to a constant, and a constant to $\top$
  - Thus, C(s, x, _) can change at most twice

Orderings

Thus the constant propagation algorithm is linear in program size

Number of steps =

Number of C(….) values computed * 2 =

Number of program statements * 4