Partial Ordering & Hasse Diagrams

•••

Gavin, Pushpdeep

Our Problem

- Survey three programming languages (Java, Python, C++)
  - Design a compare function that bestows a partial ordering on the power set with 6 elements
  - Visualize the partial ordering as a Hasse diagram

- Background information
  - Let's say our set is U = {1, 2, 3, 4, 5, 6}
  - The power set P(U) is the set of all possible subsets of U

First Language: Java

- In Java, Comparator and List.sort both expect functions to return a -/0/+ integer
- In the case of a partial ordering, where this is not possible, a backup must be used
  - We must come up with tie-breakers to restore “total order”
- We used cardinality followed by numeric mask to help achieve this goal

Second Language: Python

- Python handles this problem differently on a fundamental level
- Instead of using comparators, Python allows sorting by keys
  - Python: sorted(items, key=example_func)
  - Java: Collections.sort(list, comparator)
- Keys sidestep the requirement that the original objects are totally ordered
  - Python asks “Is key(A) < key(B)?” instead of “Is A < B?”, making everything easier

Third Language: C++

- C++ is a language that shares qualities of both Java and Python
  - It sits closer to Java in terms of ordering requirements
  - It sits closer to Python in terms of how much the language actually enforces correctness
- Sorting in C++ is mostly handled by std::sort
  - This function does not check if the requirements are broken, just like in Python
  - The difference here being that C++ still requires the rules to be followed, Python does not

Hasse Diagram

- Definition: a Hasse diagram is a graphical way to represent a finite partially ordered set by showing only the essential ordering relationships, without redundancy.
- Meaning: we are able to use Hasse diagrams to visualize how every set in P(U) is connected. These connections are layered on top of each-other, like a tree. It starts with the empty set {} and concludes with the original set, say {1,2,3,4,5,6}.