Extra Notes: Prerequisites to CRDT

Theodore Leebrant

2 March 2020

Definition 1. An **upper bound** of a subset S of a partially ordered set $(P, \leq)$ is an element $b \in P$ such that $\forall x \in S, x \leq b$.

Definition 2. An upper bound b of S is called the **least upper bound** (or supremum) of S if for all $z \in P$ where z is an upper bound of S, we have $b \leq z$.

Definition 3. Given a nonempty set S, a partial order $(S, \leq)$ and a non-empty subset T (where $T \subseteq S$), the **join** of T, is the least upper bound of $T \subseteq S$.
Let $a \vee b$ denote the join of $\{a, b\} \subseteq S$ where $a, b \in S$.

You may use the fact that if $a \vee b$ and $b \vee c$ exists, then it fulfills the associativity law, i.e. $(a \vee b) \vee c = a \vee (b \vee c)$.

Exercise 1.

1. Show that join is commutative.

2. Show that join is idempotent.

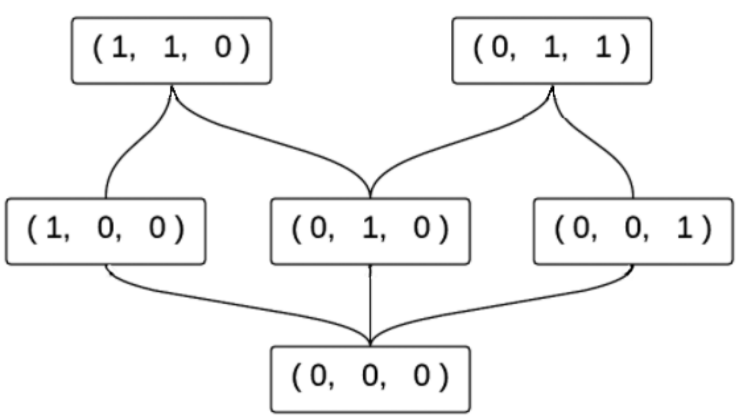
Definition 4. A **join-semilattice** is a partially ordered set that has a join for any nonempty finite subset. Equivalently, a join-semilattice is a partially ordered set S where $\forall a, b \in S, a \vee b$ exists.

Exercise 2.

1. Give a set with cardinality 2 that is a join-semilattice.

2. Give a set with cardinality 2 that is not a join-semilattice.

3. Define set $T = \{0, 1\}^3$, where the elements of T are expressed as a row vector; i.e. given that a, b, c take values either 0 or 1, $(a, b, c) \in T$. Let the partial order $\preceq$ defined on T be $(a, b, c) \preceq (\alpha, \beta, \gamma)$ iff $a \leq \alpha$, $b \leq \beta$, and $c \leq \gamma$ where the $\leq$ operator is defined as in $\mathbb{Z}$. Consider the following Hasse diagram of a subset S of T. Is this a join-semilattice? If yes, explain. If not, would it be possible to make it a join-semilattice by adding element(s) from T into the set? Explain.



This is a small exercise in using new definition not covered in class. The lead-up as well as the inspiration for this extra notes is the *Conflict-free Replicated Data Types*, which are data structures that can be updated concurrently without any locking or coordination and still remains consistent. For an easy introduction to CRDT which just uses the definition above, check out this link, or this link. For a more complete view of CRDT, you can check out the papers here.