```
┌──────────────────────── MODULE CC ────────────────────────┐
  TLA+ specification of Causal Consistency variants, including CC, CM, and CCv.

  See the paper "On Verifying Causal Consistency" (POPL'2017).
EXTENDS Naturals, Sequences, Functions, FiniteSets, FiniteSetsExt, RelationUtils, TLC

CONSTANTS Keys, Vals
InitVal ≜ CHOOSE v : v ∉ (Keys ∪ Vals)

  oid: unique operation identifier
Operation ≜ [op : {"read", "write"}, key : Keys, val : Vals, oid : Nat]
R(k, v, oid)  ≜ [op ↦ "read", key ↦ k, val ↦ v, oid ↦ oid]
W(k, v, oid)  ≜ [op ↦ "write", key ↦ k, val ↦ v, oid ↦ oid]

Session ≜ Seq(Operation)   A session s ∈ Session is a sequence of operations.
History ≜ SUBSET Session   A history h ∈ History is a set of sessions.
├───────────────────────────────────────────────────────────┤
  Utilities.
Ops(h) ≜   Return the set of all operations in history h ∈ History.
  UNION {Range(s) : s ∈ h}
├───────────────────────────────────────────────────────────┤
  Well-formedness of history h ∈ History:

  − TODO: type invariants
  - uniqueness of oids
WellFormed(h) ≜
  ∧ h ∈ History
  ∧ Cardinality(Ops(h)) = ReduceSet(LAMBDA s, x : Len(s) + x, h, 0)
├───────────────────────────────────────────────────────────┤
  Program order: a union of total orders among operations in the same session.
ProgramOrder(h) ≜
  LET RECURSIVE SessionProgramOrder(_)
      SessionProgramOrder(s) ≜
        IF s = ⟨⟩ THEN {}
         ELSE LET sh ≜ Head(s)
                  st ≜ Tail(s)
              IN  {⟨sh, t⟩ : t ∈ Range(st)} ∪ SessionProgramOrder(st)
  IN  UNION {SessionProgramOrder(s) : s ∈ h}
├───────────────────────────────────────────────────────────┤
  Sequential semantics of read-write registers.
├───────────────────────────────────────────────────────────┤
  Specification of Causal Consistency: CC, CCv, and CM
CCv(h) ≜   Check whether h ∈ History satisfies CCv (Causal Convergence)
  ∧ WellFormed(h)
  ∧ LET ops ≜ Ops(h)
    IN  ∧ ∃ co ∈ SUBSET (ops × ops) :
```

$$\exists\, arb \in \textsc{subset}\ (ops \times ops):$$
$$\land\ IsStrictPartialOrder(co,\ ops)$$
$$\land\ IsStrictTotalOrder(arb,\ ops)$$
$$\land\ Respect(co,\ ProgramOrder(h)) \quad \text{\textit{AxCausal}}$$
$$\land\ Respect(arb,\ co) \quad \text{\textit{AxArb}}$$
$$\land\ \forall\, op \in ops : \textsc{true} \quad \text{\textit{TODO}: \textit{AxCausalArb}}$$
$$\land\ \textsc{false}$$

Test case: The following histories are from Figure 2 of the *POPL*'2017 paper.

Naming Conventions:

− *ha*: history of Figure 2(*a*)
− *hasa*: session a of history ha

TODO: to automatically generate histories

$hasa \triangleq \langle W(\text{“x”}, 1, 1),\ R(\text{“x”}, 2, 2)\rangle$
$hasb \triangleq \langle W(\text{“x”}, 2, 3),\ R(\text{“x”}, 1, 4)\rangle$
$ha \triangleq \{hasa,\ hasb\}$ *CM* but not *CCv*

$hbsa \triangleq \langle W(\text{“z”}, 1, 1),\ W(\text{“x”}, 1, 2),\ W(\text{“y”}, 1, 3)\rangle$
$hbsb \triangleq \langle W(\text{“x”}, 2, 4),\ R(\text{“z”}, 0, 5),\ R(\text{“y”}, 1, 6),\ R(\text{“x”}, 2, 7)\rangle$
$hb \triangleq \{hbsa,\ hbsb\}$ *CCv* but not *CM*

$hcsa \triangleq \langle W(\text{“x”}, 1, 1)\rangle$
$hcsb \triangleq \langle W(\text{“x”}, 2, 2),\ R(\text{“x”}, 1, 3),\ R(\text{“x”}, 2, 4)\rangle$
$hc \triangleq \{hcsa,\ hcsb\}$ *CC* but not *CM* nor *CCv*

$hdsa \triangleq \langle W(\text{“x”}, 1, 1),\ R(\text{“y”}, 0, 2),\ W(\text{“y”}, 1, 3),\ R(\text{“x”}, 1, 4)\rangle$
$hdsb \triangleq \langle W(\text{“x”}, 2, 5),\ R(\text{“y”}, 0, 6),\ W(\text{“y”}, 2, 7),\ R(\text{“x”}, 2, 8)\rangle$
$hd \triangleq \{hdsa,\ hdsb\}$ *CC*, *CM*, and *CCv* but no *SC*

$hesa \triangleq \langle W(\text{“x”}, 1, 1),\ W(\text{“y”}, 1, 2)\rangle$
$hesb \triangleq \langle R(\text{“y”}, 1, 3),\ W(\text{“x”}, 2, 4)\rangle$
$hesc \triangleq \langle R(\text{“x”}, 2, 5),\ R(\text{“x”}, 1, 6)\rangle$
$he \triangleq \{hesa,\ hesb,\ hesc\}$ not *CC* (nor *CM*, nor *CCv*)

$\textsc{theorem}\ WellFormedTheorem \triangleq$
$\quad \forall\, h \in \{ha,\ hb,\ hc,\ hd,\ he\} : WellFormed(h)$

$CardOfProgramOrderOfHistory(h) \triangleq$
$\quad \textsc{let}\ CardOfProgramOrderOfSession(s) \triangleq$
$\qquad \textsc{if}\ Len(s) \leq 1\ \textsc{then}\ 0\ \textsc{else}\ Sum(1 \,..\, Len(s) - 1)$
$\quad \textsc{in}\quad ReduceSet(\textsc{lambda}\ s,\ x : CardOfProgramOrderOfSession(s) + x,\ h,\ 0)$

$\textsc{theorem}\ ProgramOrderCardinalityTheorem \triangleq$
$\quad \forall\, h \in \{ha,\ hb,\ hc,\ hd,\ he\} :$
$\qquad Cardinality(ProgramOrder(h)) = CardOfProgramOrderOfHistory(h)$

---

* Modification *History*
* Last modified Sat *Apr* 03 22:21:24 *CST* 2021 by *hengxin*
* Created *Tue Apr* 01 10:24:07 *CST* 2021 by *hengxin*