```
1  ---------- MODULE Voting ----------
2  EXTENDS Sets
3  ----------------------------------
4  CONSTANT Value, Acceptor, Quorum

6  ASSUME QuorumAssumption ≜
7      ∧  ∀ Q ∈ Quorum : Q ⊆ Acceptor
8      ∧  ∀ Q1, Q2 ∈ Quorum : Q1 ∩ Q2 ≠ {}

10 THEOREM QuorumNonEmpty ≜ ∀ Q ∈ Quorum : Q ≠ {}
11 BY QuorumAssumption

13 Ballot ≜ Nat
14 ----------------------------------
15 VARIABLES votes, maxBal

17 TypeOK ≜ ∧ votes ∈ [Acceptor → SUBSET (Ballot × Value)]
18          ∧ maxBal ∈ [Acceptor → Ballot ∪ {−1}]
19 ----------------------------------
20 VotedFor(a, b, v) ≜ ⟨b, v⟩ ∈ votes[a]

22 DidNotVoteAt(a, b) ≜ ∀ v ∈ Value : ¬ VotedFor(a, b, v)

24 ShowsSafeAt(Q, b, v) ≜
25   ∧ ∀ a ∈ Q : maxBal[a] ≥ b  have promised
26   ∧ ∃ c ∈ −1 .. (b − 1) :
27       ∧ (c ≠ −1) ⇒ ∃ a ∈ Q : VotedFor(a, c, v)
28       ∧ ∀ d ∈ (c + 1) .. (b − 1), a ∈ Q : DidNotVoteAt(a, d)
29 ----------------------------------
30 Init ≜
31     ∧ votes = [a ∈ Acceptor ↦ {}]
32     ∧ maxBal = [a ∈ Acceptor ↦ −1]

34 IncreaseMaxBal(a, b) ≜
35   ∧ b > maxBal[a]
36   ∧ maxBal′ = [maxBal EXCEPT ![a] = b]  make promise
37   ∧ UNCHANGED votes

39 VoteFor(a, b, v) ≜
40     ∧  maxBal[a] ≤ b  keep promise
41     ∧  ∀ vt ∈ votes[a] : vt[1] ≠ b
42     ∧  ∀ c ∈ Acceptor \ {a} :
43          ∀ vt ∈ votes[c] : (vt[1] = b) ⇒ (vt[2] = v)
44     ∧  ∃ Q ∈ Quorum : ShowsSafeAt(Q, b, v)  safe to vote
45     ∧  votes′ = [votes EXCEPT ![a] = votes[a] ∪ {⟨b, v⟩}]  vote
46     ∧  maxBal′ = [maxBal EXCEPT ![a] = b]  make promise
47 ----------------------------------
```

$$Next \triangleq$$
$$\quad \exists\, a \in Acceptor,\, b \in Ballot :$$
$$\quad\quad \lor\, IncreaseMaxBal(a,\, b)$$
$$\quad\quad \lor \exists\, v \in Value : VoteFor(a,\, b,\, v)$$

$$Spec \triangleq Init \land \Box[Next]_{\langle votes,\, maxBal\rangle}$$

$$ChosenAt(b,\, v) \triangleq$$
$$\quad \exists\, Q \in Quorum : \forall\, a \in Q : VotedFor(a,\, b,\, v)$$

$$chosen \triangleq \{v \in Value : \exists\, b \in Ballot : ChosenAt(b,\, v)\}$$

$$CannotVoteAt(a,\, b) \triangleq$$
$$\quad \land\, maxBal[a] > b$$
$$\quad \land\, DidNotVoteAt(a,\, b)$$

$$NoneOtherChoosableAt(b,\, v) \triangleq$$
$$\quad \exists\, Q \in Quorum :$$
$$\quad\quad \forall\, a \in Q : VotedFor(a,\, b,\, v) \lor CannotVoteAt(a,\, b)$$

$$SafeAt(b,\, v) \triangleq$$
$$\quad \forall\, c \in 0\,..\,(b-1) : NoneOtherChoosableAt(c,\, v)$$

$$VotesSafe \triangleq$$
$$\quad \forall\, a \in Acceptor,\, b \in Ballot,\, v \in Value :$$
$$\quad\quad VotedFor(a,\, b,\, v) \Rightarrow SafeAt(b,\, v)$$

$$OneVote \triangleq$$
$$\quad \forall\, a \in Acceptor,\, b \in Ballot,\, v,\, w \in Value :$$
$$\quad\quad VotedFor(a,\, b,\, v) \land VotedFor(a,\, b,\, w) \Rightarrow (v = w)$$

$$OneValuePerBallot \triangleq$$
$$\quad \forall\, a1,\, a2 \in Acceptor,\, b \in Ballot,\, v1,\, v2 \in Value :$$
$$\quad\quad VotedFor(a1,\, b,\, v1) \land VotedFor(a2,\, b,\, v2) \Rightarrow (v1 = v2)$$

$$Inv \triangleq TypeOK \land VotesSafe \land OneValuePerBallot$$

THEOREM $AllSafeAtZero \triangleq \forall\, v \in Value : SafeAt(0,\, v)$
BY DEF $SafeAt$

THEOREM $ChoosableThm \triangleq$
$$\quad\quad \forall\, b \in Ballot,\, v \in Value :$$
$$\quad\quad\quad ChosenAt(b,\, v) \Rightarrow NoneOtherChoosableAt(b,\, v)$$
BY DEF $ChosenAt$, $NoneOtherChoosableAt$

THEOREM $OneVoteThm \triangleq OneValuePerBallot \Rightarrow OneVote$
BY DEF $OneValuePerBallot$, $OneVote$

```
THEOREM VotesSafeImpliesConsistency ≜
  ASSUME VotesSafe, OneVote, chosen ≠ {}
  PROVE  ∃ v ∈ Value : chosen = {v}
⟨1⟩1. PICK v ∈ Value : v ∈ chosen
  BY  DEF chosen
⟨1⟩2. SUFFICES ASSUME NEW w ∈ chosen
                PROVE  w = v
  BY ⟨1⟩1, ⟨1⟩2
⟨1⟩3. ASSUME NEW b1 ∈ Ballot, NEW b2 ∈ Ballot, b1 < b2,
             NEW v1 ∈ Value, NEW v2 ∈ Value,
             ChosenAt(b1, v1) ∧ ChosenAt(b2, v2)
     PROVE  v1 = v2
  ⟨2⟩1. SafeAt(b2, v2)
    BY ⟨1⟩3, QuorumAssumption, SMT DEF ChosenAt, VotesSafe
  ⟨2⟩2. QED
    BY ⟨1⟩3, ⟨2⟩1, QuorumAssumption, Z3
    DEFS CannotVoteAt, DidNotVoteAt, OneVote,
         ChosenAt, NoneOtherChoosableAt, Ballot, SafeAt
⟨1⟩4. QED
  BY QuorumAssumption, ⟨1⟩1, ⟨1⟩2, ⟨1⟩3, Z3
  DEFS Ballot, ChosenAt, OneVote, chosen

THEOREM ShowsSafety ≜
          TypeOK ∧ VotesSafe ∧ OneValuePerBallot ⇒
            ∀ Q ∈ Quorum, b ∈ Ballot, v ∈ Value :
              ShowsSafeAt(Q, b, v) ⇒ SafeAt(b, v)
  BY QuorumAssumption, Z3
  DEFS Ballot, TypeOK, VotesSafe, OneValuePerBallot, SafeAt,
    ShowsSafeAt, CannotVoteAt, NoneOtherChoosableAt, DidNotVoteAt
```

```
THEOREM Invariance ≜ Spec ⇒ □Inv
⟨1⟩ USE  DEF Inv
⟨1⟩1. Init ⇒ Inv
  BY  DEF Init, TypeOK, VotesSafe, OneValuePerBallot, VotedFor
⟨1⟩2. Inv ∧ [Next]⟨votes, maxBal⟩ ⇒ Inv′
  ⟨2⟩ SUFFICES ASSUME Inv, [Next]⟨votes, maxBal⟩
               PROVE  Inv′
    OBVIOUS
  ⟨2⟩1.CASE Next
    ⟨3⟩ SUFFICES ASSUME NEW a ∈ Acceptor, NEW b ∈ Ballot,
                        ∨ IncreaseMaxBal(a, b)
                        ∨ ∃ v ∈ Value : VoteFor(a, b, v)
                 PROVE  Inv′
      BY ⟨2⟩1  DEF Next
    ⟨3⟩1.CASE IncreaseMaxBal(a, b)
```

⟨4⟩1. $TypeOK'$
BY ⟨3⟩1 DEF $TypeOK$, $IncreaseMaxBal$
⟨4⟩2. $VotesSafe'$
⟨5⟩ SUFFICES ASSUME NEW $a\_1 \in Acceptor'$, NEW $b\_1 \in Ballot'$, NEW $v \in Value'$
PROVE $VotedFor(a\_1,\ b\_1,\ v)' \Rightarrow SafeAt(b\_1,\ v)'$
BY DEF $VotesSafe$
⟨5⟩1. $\forall\, aa \in Acceptor,\ bb \in Ballot,\ vv \in Value :$
$VotedFor(aa,\ bb,\ vv) \equiv VotedFor(aa,\ bb,\ vv)'$
BY ⟨3⟩1 DEF $IncreaseMaxBal$, $VotedFor$
⟨5⟩2. $\forall\, aa \in Acceptor,\ bb \in Ballot :$
$maxBal[aa] > bb \Rightarrow maxBal'[aa] > bb$
BY ⟨3⟩1 DEF $IncreaseMaxBal$, $TypeOK$, $Ballot$
⟨5⟩3. $\forall\, aa \in Acceptor,\ bb \in Ballot :$
$DidNotVoteAt(aa,\ bb) \Rightarrow DidNotVoteAt(aa,\ bb)'$
BY ⟨3⟩1 DEF $IncreaseMaxBal$, $DidNotVoteAt$, $VotedFor$
⟨5⟩4. $\forall\, aa \in Acceptor,\ bb \in Ballot :$
$CannotVoteAt(aa,\ bb) \Rightarrow CannotVoteAt(aa,\ bb)'$
BY ⟨3⟩1, ⟨5⟩2, ⟨5⟩3 DEF $IncreaseMaxBal$, $CannotVoteAt$
⟨5⟩5. $\forall\, bb \in Ballot,\ vv \in Value :$
$NoneOtherChoosableAt(bb,\ vv) \Rightarrow NoneOtherChoosableAt(bb,\ vv)'$
BY ⟨5⟩1, ⟨5⟩4, $QuorumAssumption$DEFS $NoneOtherChoosableAt$
⟨5⟩6. QED
BY ⟨5⟩1, ⟨5⟩5 DEF $TypeOK$, $Ballot$, $VotesSafe$, $SafeAt$
⟨4⟩3. $OneValuePerBallot'$
BY ⟨3⟩1 DEF $IncreaseMaxBal$, $OneValuePerBallot$, $VotedFor$
⟨4⟩4. QED
BY ⟨4⟩1, ⟨4⟩2, ⟨4⟩3 DEF Inv
⟨3⟩2. ASSUME NEW $v \in Value$,
$VoteFor(a,\ b,\ v)$
PROVE Inv'
⟨4⟩ SUFFICES ASSUME NEW $Q \in Quorum$,
$ShowsSafeAt(Q,\ b,\ v)$
PROVE Inv'
BY ⟨3⟩2 DEF $VoteFor$
⟨4⟩1. $TypeOK'$
BY ⟨3⟩2 DEF $TypeOK$, $VoteFor$
⟨4⟩2. $VotesSafe'$ Using $OneValuePerBallot$?
BY ⟨3⟩2, $ShowsSafety$, $QuorumAssumption$ DEFS $Ballot$, $VoteFor$, $VotesSafe$, $SafeAt$, $ShowsSafeAt$, $CannotVoteAt$, $NoneOtherChoosableAt$, $DidNotVoteAt$, $VotedFor$, $OneValuePerBallot$
⟨4⟩3. $OneValuePerBallot'$
BY ⟨3⟩2 DEF $VoteFor$, $OneValuePerBallot$, $VotedFor$, $TypeOK$
⟨4⟩4. QED
BY ⟨3⟩2, ⟨4⟩1, ⟨4⟩2, ⟨4⟩3 DEF Inv
⟨3⟩3. QED

```
        BY ⟨2⟩1, ⟨3⟩1, ⟨3⟩2
    ⟨2⟩2.CASE UNCHANGED ⟨votes, maxBal⟩
      BY ⟨2⟩2
      DEFS TypeOK, Next, VotesSafe, OneValuePerBallot,
           VotedFor, SafeAt, NoneOtherChoosableAt, CannotVoteAt, DidNotVoteAt,
           IncreaseMaxBal, VoteFor
    ⟨2⟩3. QED
      BY ⟨2⟩1, ⟨2⟩2
  ⟨1⟩3. QED
    BY ⟨1⟩1, ⟨1⟩2, PTL DEF Spec
──────────────────────────────────────────────────────────────────────────────
  C ≜ INSTANCE Consensus

  THEOREM Spec ∧ Inv ⇒ C!Spec
  ⟨1⟩1. Init ⇒ C!Init
    BY QuorumAssumption, SetExtensionality, IsaM("force")
     DEF Init, C!Init, chosen, ChosenAt, VotedFor
  ⟨1⟩2. Next ∧ Inv ⇒ C!Next ∨ UNCHANGED chosen
    ⟨2⟩1 SUFFICES ASSUME Next, InvPROVE C!Next ∨ UNCHANGED chosen
      BY ⟨2⟩1
    ⟨2⟩2. chosen ⊆ chosen′
      BY ⟨2⟩1, QuorumAssumption, Z3   SMTT(10) fails
       DEF Next, Inv, TypeOK, IncreaseMaxBal, chosen, ChosenAt, VotedFor, Ballot, VoteFor
    ⟨2⟩3. chosen′ = {} ∨ ∃ v ∈ Value : chosen′ = {v}
      ⟨3⟩1. PICK a ∈ Acceptor, b ∈ Ballot :
                 ∨ IncreaseMaxBal(a, b)
                 ∨ ∃ v ∈ Value : VoteFor(a, b, v)
        BY ⟨2⟩1  DEF Next
      ⟨3⟩2.CASE IncreaseMaxBal(a, b)
      ⟨3⟩3.CASE ∃ v ∈ Value : VoteFor(a, b, v)
      ⟨3⟩q. QED
        BY ⟨3⟩1, ⟨3⟩2, ⟨3⟩3, SMT
    ⟨2⟩q. QED
      BY ⟨2⟩1, ⟨2⟩2, ⟨2⟩3, OneVoteThm, VotesSafeImpliesConsistency, SetExtensionality, SMT
       DEF Inv, C!Next
  ⟨1⟩3. QED
    PROOF OMITTED
└──────────────────────────────────────────────────────────────────────────────┘
```