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

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

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

35 IncreaseMaxBal(a, b) ≜
36   ∧ b > maxBal[a]
37   ∧ maxBal' = [maxBal EXCEPT ![a] = b]
38   ∧ UNCHANGED votes

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

```
────────────────────────────────────────────────────────────
Next ≜
    ∃ a ∈ Acceptor, b ∈ Ballot :
        ∨ IncreaseMaxBal(a, b)
        ∨ ∃ v ∈ Value : VoteFor(a, b, v)

Spec ≜ Init ∧ □[Next]_⟨votes, maxBal⟩
────────────────────────────────────────────────────────────
ChosenAt(b, v) ≜
    ∃ Q ∈ Quorum :
        ∀ a ∈ Q : VotedFor(a, b, v)

chosen ≜ {v ∈ Value : ∃ b ∈ Ballot : ChosenAt(b, v)}
────────────────────────────────────────────────────────────
CannotVoteAt(a, b) ≜
    ∧ maxBal[a] > b
    ∧ DidNotVoteAt(a, b)

NoneOtherChoosableAt(b, v) ≜
    ∃ Q ∈ Quorum :
        ∀ a ∈ Q : VotedFor(a, b, v) ∨ CannotVoteAt(a, b)

SafeAt(b, v) ≜
    ∀ c ∈ 0 . . (b − 1) :
        NoneOtherChoosableAt(c, v)

VotesSafe ≜
    ∀ a ∈ Acceptor, b ∈ Ballot, v ∈ Value :
        VotedFor(a, b, v) ⇒ SafeAt(b, v)

OneVote ≜
    ∀ a ∈ Acceptor, b ∈ Ballot, v, w ∈ Value :
        VotedFor(a, b, v) ∧ VotedFor(a, b, w) ⇒ (v = w)
OneValuePerBallot ≜
    ∀ a1, a2 ∈ Acceptor, b ∈ Ballot, v1, v2 ∈ Value :
        VotedFor(a1, b, v1) ∧ VotedFor(a2, b, v2) ⇒ (v1 = v2)

Inv ≜ TypeOK ∧ VotesSafe ∧ OneValuePerBallot
────────────────────────────────────────────────────────────
THEOREM AllSafeAtZero ≜ ∀ v ∈ Value : SafeAt(0, v)
  BY DEF SafeAt

THEOREM ChoosableThm ≜
            ∀ b ∈ Ballot, v ∈ Value :
              ChosenAt(b, v) ⇒ NoneOtherChoosableAt(b, v)
  BY DEF ChosenAt, NoneOtherChoosableAt

THEOREM OneVoteThm ≜ OneValuePerBallot ⇒ OneVote
```

```
96    BY DEF OneValuePerBallot, OneVote
97  ⊢------------------------------------------------------------------⊣
98  THEOREM VotesSafeImpliesConsistency ≜
99    ASSUME VotesSafe, OneVote, chosen ≠ {}
100   PROVE  ∃ v ∈ Value : chosen = {v}
101 ⟨1⟩1. PICK v ∈ Value : v ∈ chosen
    ⟨1⟩1. SUFFICES ASSUME NEW v ∈ Value, v ∈ chosen PROVE ∃ u ∈ Value : chosen = {u}
106   BY DEF chosen
107 ⟨1⟩2. SUFFICES ASSUME NEW w ∈ chosen
108                PROVE  w = v
109   BY ⟨1⟩1, ⟨1⟩2
110 ⟨1⟩3. ASSUME NEW b1 ∈ Ballot, NEW b2 ∈ Ballot, b1 < b2,
111                NEW v1 ∈ Value, NEW v2 ∈ Value,
112                ChosenAt(b1, v1) ∧ ChosenAt(b2, v2)
113       PROVE  v1 = v2
114   ⟨2⟩1. SafeAt(b2, v2)
115     BY ⟨1⟩3, QuorumAssumption, SMT DEF ChosenAt, VotesSafe
116   ⟨2⟩2. QED
117     BY ⟨1⟩3, ⟨2⟩1, QuorumAssumption, Z3
118     DEFS CannotVoteAt, DidNotVoteAt, OneVote,
119          ChosenAt, NoneOtherChoosableAt, Ballot, SafeAt
120 ⟨1⟩4. QED
121   BY QuorumAssumption, ⟨1⟩1, ⟨1⟩2, ⟨1⟩3, Z3
122   DEFS Ballot, ChosenAt, OneVote, chosen

124 THEOREM ShowsSafety ≜
125              TypeOK ∧ VotesSafe ∧ OneValuePerBallot ⇒
126                ∀ Q ∈ Quorum, b ∈ Ballot, v ∈ Value :
127                  ShowsSafeAt(Q, b, v) ⇒ SafeAt(b, v)
128   BY QuorumAssumption, Z3
129   DEFS Ballot, TypeOK, VotesSafe, OneValuePerBallot, SafeAt,
130     ShowsSafeAt, CannotVoteAt, NoneOtherChoosableAt, DidNotVoteAt
131 ⊢------------------------------------------------------------------⊣
132 THEOREM Invariance ≜ Spec ⇒ □Inv
133 ⟨1⟩1. Init ⇒ Inv
134   BY SMT DEF Init, Inv, VotesSafe, VotedFor, TypeOK, VotesSafe, OneValuePerBallot
135 ⟨1⟩2. ASSUME Inv, [Next]⟨votes, maxBal⟩
136       PROVE  Inv′
137   ⟨2⟩ USE DEF Inv, Ballot, VotedFor, VoteFor
138   ⟨2⟩1.CASE UNCHANGED ⟨votes, maxBal⟩
139     BY ⟨1⟩2, ⟨2⟩1, IsaM("auto")
140     DEFS IncreaseMaxBal, ShowsSafeAt,
141          DidNotVoteAt, TypeOK, VotesSafe, OneValuePerBallot,
142          SafeAt, NoneOtherChoosableAt, CannotVoteAt
143   ⟨2⟩2.CASE Next
```

```
144   ⟨2⟩3. QED
145     BY ⟨2⟩1, ⟨2⟩2, ⟨1⟩2, SMT
146  ⟨1⟩3. QED
147    BY ⟨1⟩1, ⟨1⟩2, PTL DEF Spec
148 ├──────────────────────────────────────────────────────────────────────────┤
149  C ≜ INSTANCE Consensus

151  THEOREM Spec ∧ Inv ⇒ C!Spec
152  ⟨1⟩1. Init ⇒ C!Init
153    BY QuorumAssumption, SetExtensionality, IsaM("force")
154     DEF Init, C!Init, chosen, ChosenAt, VotedFor
155  ⟨1⟩2. Next ∧ Inv ⇒ C!Next ∨ UNCHANGED chosen
156    ⟨2⟩1 SUFFICES ASSUME Next, InvPROVE C!Next ∨ UNCHANGED chosen
157      BY ⟨2⟩1
158    ⟨2⟩2. chosen ⊆ chosen′
159      BY ⟨2⟩1, QuorumAssumption, Z3   SMTT(10) fails
160       DEF Next, Inv, TypeOK, IncreaseMaxBal, chosen, ChosenAt, VotedFor, Ballot, VoteFor
161    ⟨2⟩3. chosen′ = {} ∨ ∃ v ∈ Value : chosen′ = {v}
162      ⟨3⟩1. PICK a ∈ Acceptor, b ∈ Ballot :
163                ∨ IncreaseMaxBal(a, b)
164                ∨ ∃ v ∈ Value : VoteFor(a, b, v)
165        BY ⟨2⟩1  DEF Next
166      ⟨3⟩2.CASE IncreaseMaxBal(a, b)
167      ⟨3⟩3.CASE ∃ v ∈ Value : VoteFor(a, b, v)
168      ⟨3⟩q. QED
169        BY ⟨3⟩1, ⟨3⟩2, ⟨3⟩3, SMT
170    ⟨2⟩q. QED
171      BY ⟨2⟩1, ⟨2⟩2, ⟨2⟩3, OneVoteThm, VotesSafeImpliesConsistency, SetExtensionality, SMT
172       DEF Inv, C!Next
173  ⟨1⟩3. QED
174    PROOF OMITTED
175 └──────────────────────────────────────────────────────────────────────────┘
```