```
- MODULE Voting -
 1 [
    This is a high-level algorithm in which a set of processes cooperatively choose a value.
    EXTENDS Integers
 7 |
    CONSTANTS
          Value.
                         The set of choosable values.
         Acceptor,
                         A set of processes that will choose a value.
10
          Quorum
                         The set of "quorums", where a "quorum" is a "large enough" set of acceptors.
11
    Assume QuorumAssumption \triangleq
13
          \land \quad \forall \ Q \in Quorum : Q \subseteq Acceptor
14
              \forall Q1, Q2 \in Quorum : Q1 \cap Q2 \neq \{\}
15
    THEOREM QuorumNonEmpty \triangleq \forall Q \in Quorum : Q \neq \{\}
17
    Ballot \stackrel{\Delta}{=} Nat The set of "ballot numbers".
19
20 |
    Each acceptor can cast one or more votes, where each vote cast by an acceptor has the form \langle b, v \rangle
    indicating that the acceptor has voted for value v in ballot b.
26
    VARIABLES
          votes,
                      votes[a]: the set of votes cast by acceptor a
27
         maxBal
                      maxBal[a]: a ballot number.
28
                      Acceptor a will cast further votes only in ballots numbered > maxBal[a].
29
     TypeOK \triangleq
31
          \land votes \in [Acceptor \rightarrow SUBSET (Ballot \times Value)]
32
          \land maxBal \in [Acceptor \rightarrow Ballot \cup \{-1\}]
33
34
     VotedFor(a, b, v) \stackrel{\Delta}{=} \langle b, v \rangle \in votes[a] Acceptor a has voted for v in ballot b.
35
     ChosenAt(b, v) \stackrel{\Delta}{=} \langle b, v \rangle is chosen if a quorum of acceptors have voted for it.
37
         \exists Q \in Quorum :
38
            \forall a \in Q : VotedFor(a, b, v)
39
     chosen \stackrel{\Delta}{=} The set of values that have been chosen.
41
         \{v \in Value : \exists b \in Ballot : ChosenAt(b, v)\}
42
43
     DidNotVoteAt(a, b) \stackrel{\triangle}{=} The acceptor a did not vote (for any value) at ballot b.
44
         \forall v \in Value : \neg VotedFor(a, b, v)
45
     Cannot VoteAt(a, b) \stackrel{\triangle}{=} The acceptor a cannot vote (for any value) at ballot b.
47
          \wedge DidNotVoteAt(a, b)
48
          \wedge maxBal[a] > b
49
    NoneOtherChoosableAt(b, v) \stackrel{\Delta}{=} ChosenAt(b, w) is not and can never become true for any w \neq v.
51
         \exists Q \in Quorum :
52
            \forall a \in Q : VotedFor(a, b, v) \lor CannotVoteAt(a, b)
```

```
THEOREM ChoosableThm \stackrel{\triangle}{=}
                      \forall b \in Ballot, v \in Value:
 56
                         ChosenAt(b, v) \Rightarrow NoneOtherChoosableAt(b, v)
 57
 58
     SafeAt(b, v) \triangleq
                             No value other than v has been or can ever be chosen in any ballot < b.
 59
          \forall c \in 0 ... (b-1) : NoneOtherChoosableAt(c, v)
 60
     THEOREM AllSafeAtZero \stackrel{\Delta}{=} \forall v \in Value : SafeAt(0, v)
 62
 63
      One Vote \stackrel{\triangle}{=} \forall a \in Acceptor, b \in Ballot, v, w \in Value:
 64
                          VotedFor(a, b, v) \land VotedFor(a, b, w) \Rightarrow (v = w)
 65
      One Value Per Ballot \triangleq
 67
          \forall a1, a2 \in Acceptor, b \in Ballot, v1, v2 \in Value :
 68
             VotedFor(a1, b, v1) \land VotedFor(a2, b, v2) \Rightarrow (v1 = v2)
 69
     THEOREM OneValuePerBallot \Rightarrow OneVote
 71
 72
      VotesSafe \stackrel{\triangle}{=} \forall a \in Acceptor, b \in Ballot, v \in Value :
 73
                            VotedFor(a, b, v) \Rightarrow SafeAt(b, v)
 74
     THEOREM VotesSafeImpliesConsistency \stackrel{\Delta}{=}
 76
                       \land TypeOK
 77
                       \land VotesSafe
 78
                       \land OneVote
 79
                       \Rightarrow \lor chosen = \{\}
 80
                            \lor \exists v \in Value : chosen = \{v\}
 81
 82
      ShowsSafeAt(Q, b, v) \stackrel{\Delta}{=}
 83
           \land \forall a \in Q : maxBal[a] \ge b
 84
           \wedge \exists c \in -1 \dots (b-1):
 85
                \land (c \neq -1) \Rightarrow \exists a \in Q : VotedFor(a, c, v)
 86
                \land \forall d \in (c+1) \dots (b-1), a \in Q : DidNotVoteAt(a, d)
 87
      THEOREM ShowsSafety \stackrel{\Delta}{=}
 89
                       TypeOK \land VotesSafe \land OneValuePerBallot \Rightarrow
 90
                          \forall Q \in Quorum, b \in Ballot, v \in Value:
 91
 92
                             ShowsSafeAt(Q, b, v) \Rightarrow SafeAt(b, v)
 93
     Init \triangleq
 94
           \land votes = [a \in Acceptor \mapsto \{\}]
 95
           \land maxBal = [a \in Acceptor \mapsto -1]
 96
      IncreaseMaxBal(a, b) \stackrel{\Delta}{=} Acceptor a is allowed of increase maxBal[a] to a ballot number.
 98
           \wedge b > maxBal[a]
 99
           \wedge maxBal' = [maxBal \ EXCEPT \ ![a] = b]
100
           \land UNCHANGED votes
101
```

```
VoteFor(a, b, v) \stackrel{\Delta}{=} Acceptor a votes for v in ballot b.
103
              \land maxBal[a] \leq \overline{b} The acceptor cannot cast a vote in a ballot less than maxBal[a]
104
                  \forall vt \in votes[a] : vt[1] \neq b to maintain OneValuePerBallot
105
                   \forall c \in Acceptor \setminus \{a\}:
106
                       \forall \, vt \in votes[c] : (vt[1] = b) \Rightarrow (vt[2] = v)
107
                  \exists Q \in Quorum : ShowsSafeAt(Q, b, v) to maintain VotesSafe
108
                   votes' = [votes \ \text{EXCEPT} \ ![a] = @ \cup \{\langle b, v \rangle\}]
109
                    maxBal' = [maxBal \text{ EXCEPT } ![a] = b]
110
111 |
      Next \triangleq
112
             \exists a \in Acceptor, b \in Ballot :
113
                  \vee IncreaseMaxBal(a, b)
114
                  \forall \exists v \in Value : VoteFor(a, b, v)
115
       Spec \stackrel{\Delta}{=} Init \wedge \Box [Next]_{\langle votes, \, maxBal \rangle}
      Inv \stackrel{\triangle}{=} TypeOK \wedge VotesSafe \wedge OneValuePerBallot
      THEOREM Invariance \stackrel{\triangle}{=} Spec \Rightarrow \Box Inv
121
       C \stackrel{\Delta}{=} \text{INSTANCE } Consensus \quad \text{with } Value \leftarrow Value, chosen \leftarrow chosen
123
      THEOREM Spec \Rightarrow C!Spec
       \langle 1 \rangle 1. Inv \wedge Init \Rightarrow C!Init
126
       \langle 1 \rangle 2. \ Inv \wedge [Next]_{\langle votes, \ maxBal \rangle} \Rightarrow [C! Next]_{chosen}
127
128
          \langle 2 \rangle 1.\Box Inv \wedge \Box [Next]_{\langle votes, \, maxBal \rangle} \Rightarrow \Box [C!Next]_{chosen}
129
            BY \langle 1 \rangle 2 and temporal reasoning
130
          \langle 2 \rangle 2. \Box Inv \wedge Spec \Rightarrow C! Spec
131
            BY \langle 2 \rangle 1, \langle 1 \rangle 1
132
          \langle 2 \rangle 3. QED
133
             By \langle 2 \rangle 2, Invariance
134
135 L
```