

Users vote for a 1 .. 3 Movies during one week. After the week, the most voted Movie is displayed.

EXTENDS *Integers, FiniteSets*

$PT \triangleq$ INSTANCE *PT*

CONSTANTS *MOVIES, VOTE_DURATION, USERS*

ASSUME $MOVIES \subseteq Int$

ASSUME $\forall m \in MOVIES : m \geq 0$

ASSUME $USERS \subseteq Int$

ASSUME $VOTE_DURATION \in Int$

ASSUME $VOTE_DURATION > 0$

$MoviesVotes \triangleq$
 $[MOVIES \rightarrow \text{SUBSET } USERS]$

--algorithm *ElectionOfMovie*

variables

$days_left \in 1 .. VOTE_DURATION,$
 $movies_votes = [m \in MOVIES \mapsto \{\}],$
 $winner = -1;$

define

$TotalVoted \triangleq$
 $PT!ReduceSet(\$
 $\quad \text{LAMBDA } m, acc : acc + Cardinality(movies_votes[m]),$
 $\quad \text{DOMAIN } movies_votes, 0)$

$TotalVotedOf(movie) \triangleq$
 $Cardinality(movies_votes[movie])$

$HasMovieToVote(user) \triangleq$
 $\exists m \in MOVIES : user \notin movies_votes[m]$

$ChooseNotVotedMovies(n, user) \triangleq$
 $PT!ReduceSet(\$
 $\quad \text{LAMBDA } i, acc : \text{IF } HasMovieToVote(user)$
 $\quad \quad \text{THEN } acc \cup \{$
 $\quad \quad \quad \text{CHOOSE } m \in MOVIES :$
 $\quad \quad \quad \quad m \notin acc \wedge user \notin movies_votes[m]\}$
 $\quad \quad \text{ELSE } acc,$
 $\quad 1 .. n, \{\})$

$NumberOfMoviesToVote(user) \triangleq$
 $Cardinality(PT!ReduceSet(\$
 $\quad \text{LAMBDA } m, acc : \text{IF } user \notin movies_votes[m]$

```

        THEN  $acc \cup \{m\}$ 
        ELSE  $acc,$ 
        MOVIES,  $\{\}$ ))

Winner  $\triangleq$  CHOOSE  $m \in MOVIES : \forall m2 \in (MOVIES \setminus \{m\}) :$ 
    TotalVotedOf( $m$ )  $\geq$  TotalVotedOf( $m2$ )
end define ;

fair process User  $\in USERS$ 
variable votesLeft = 3 ;
begin Loop:
    while days_left > 0  $\wedge$  votesLeft > 0
    do
        with numberOfVotes  $\in 1 \dots PT!Min(NumberOfMoviesToVote(self), votesLeft),$ 
            moviesToVote = ChooseNotVotedMovies(numberOfVotes, self)
        do
            movies_votes := PT!ReduceSet(
                LAMBDA  $m, acc : [acc \text{ EXCEPT}$ 
                     $![m] = acc[m] \cup \{self\}],$ 
                moviesToVote, movies_votes) ;
            votesLeft := votesLeft - numberOfVotes ;
        end with ;

        days_left := days_left - 1 ;
    end while ;

    DisplayWinner:
        winner := Winner ;
end process ;
end algorithm ;

BEGIN TRANSLATION ( $chksum(pcal) = \text{"7294aa2a"} \wedge chksum(tla) = \text{"dbec08ab"}$ )
VARIABLES days_left, movies_votes, winner, pc

define statement
TotalVoted  $\triangleq$ 
    PT!ReduceSet(
        LAMBDA  $m, acc : acc + Cardinality(movies\_votes[m]),$ 
        DOMAIN movies_votes, 0)

TotalVotedOf(movie)  $\triangleq$ 
    Cardinality(movies_votes[movie])

HasMovieToVote(user)  $\triangleq$ 
     $\exists m \in MOVIES : user \notin movies\_votes[m]$ 

ChooseNotVotedMovies( $n, user$ )  $\triangleq$ 
    PT!ReduceSet(

```

```

    LAMBDA  $i, acc$  : IF  $HasMovieToVote(user)$ 
    THEN  $acc \cup \{$ 
      CHOOSE  $m \in MOVIES$  :
         $m \notin acc \wedge user \notin movies\_votes[m]\}$ 
    ELSE  $acc,$ 
    1 ..  $n, \{\}$ )

 $NumberOfMoviesToVote(user) \triangleq$ 
   $Cardinality(PT!ReduceSet($ 
    LAMBDA  $m, acc$  : IF  $user \notin movies\_votes[m]$ 
    THEN  $acc \cup \{m\}$ 
    ELSE  $acc,$ 
     $MOVIES, \{\})$ )

 $Winner \triangleq$  CHOOSE  $m \in MOVIES : \forall m2 \in (MOVIES \setminus \{m\}) :$ 
   $TotalVotedOf(m) \geq TotalVotedOf(m2)$ 

VARIABLE  $votesLeft$ 

 $vars \triangleq \langle days\_left, movies\_votes, winner, pc, votesLeft \rangle$ 

 $ProcSet \triangleq (USERS)$ 

 $Init \triangleq$  Global variables
   $\wedge days\_left \in 1 \dots VOTE\_DURATION$ 
   $\wedge movies\_votes = [m \in MOVIES \mapsto \{\}]$ 
   $\wedge winner = -1$ 
  Process User
   $\wedge votesLeft = [self \in USERS \mapsto 3]$ 
   $\wedge pc = [self \in ProcSet \mapsto \text{"Loop"}]$ 

 $Loop(self) \triangleq$   $\wedge pc[self] = \text{"Loop"}$ 
   $\wedge$  IF  $days\_left > 0 \wedge votesLeft[self] > 0$ 
    THEN  $\wedge \exists numberOfVotes \in 1 \dots PT!Min(NumberOfMoviesToVote(self), votesLeft[self])$ 
      LET  $moviesToVote \triangleq ChooseNotVotedMovies(numberOfVotes, self)$  IN
       $\wedge movies\_votes' = PT!ReduceSet($ 
        LAMBDA  $m, acc$  :  $[acc \text{ EXCEPT } ! [m] = acc[m] \cup \{self\}],$ 
         $moviesToVote, movies\_votes)$ 
       $\wedge votesLeft' = [votesLeft \text{ EXCEPT } ! [self] = votesLeft[self] - numberOfVotes]$ 
       $\wedge days\_left' = days\_left - 1$ 
       $\wedge pc' = [pc \text{ EXCEPT } ! [self] = \text{"Loop"}]$ 
    ELSE  $\wedge pc' = [pc \text{ EXCEPT } ! [self] = \text{"DisplayWinner"}]$ 
       $\wedge$  UNCHANGED  $\langle days\_left, movies\_votes, votesLeft \rangle$ 
       $\wedge$  UNCHANGED  $winner$ 

 $DisplayWinner(self) \triangleq \wedge pc[self] = \text{"DisplayWinner"}$ 

```

$$\begin{aligned}
& \wedge \text{winner}' = \text{Winner} \\
& \wedge \text{pc}' = [\text{pc} \text{ EXCEPT } ![\text{self}] = \text{"Done"}] \\
& \wedge \text{UNCHANGED } \langle \text{days_left}, \text{movies_votes}, \text{votesLeft} \rangle
\end{aligned}$$

$$\text{User}(\text{self}) \triangleq \text{Loop}(\text{self}) \vee \text{DisplayWinner}(\text{self})$$

Allow infinite stuttering to prevent deadlock on termination.

$$\begin{aligned}
\text{Terminating} \triangleq & \wedge \forall \text{self} \in \text{ProcSet} : \text{pc}[\text{self}] = \text{"Done"} \\
& \wedge \text{UNCHANGED } \text{vars}
\end{aligned}$$

$$\begin{aligned}
\text{Next} \triangleq & (\exists \text{self} \in \text{USERS} : \text{User}(\text{self})) \\
& \vee \text{Terminating}
\end{aligned}$$

$$\begin{aligned}
\text{Spec} \triangleq & \wedge \text{Init} \wedge \Box[\text{Next}]_{\text{vars}} \\
& \wedge \forall \text{self} \in \text{USERS} : \text{WF}_{\text{vars}}(\text{User}(\text{self}))
\end{aligned}$$

$$\text{Termination} \triangleq \Diamond(\forall \text{self} \in \text{ProcSet} : \text{pc}[\text{self}] = \text{"Done"})$$

END TRANSLATION

$$\begin{aligned}
\text{SumVotesLeft} \triangleq & \\
& \text{PT!ReduceSeq}(\\
& \quad \text{LAMBDA } i, \text{acc} : \text{acc} + i, \\
& \quad \text{votesLeft}, 0)
\end{aligned}$$

$$\begin{aligned}
\text{TypeOK} \triangleq & \\
& \wedge \text{movies_votes} \in \text{MoviesVotes} \\
& \wedge \text{days_left} \in 0 \dots \text{VOTE_DURATION} \\
& \wedge \forall u \in \text{USERS} : \text{votesLeft}[u] \in 0 \dots 3 \\
& \wedge \text{winner} = -1 \vee \text{winner} \in \text{MOVIES}
\end{aligned}$$

$$\begin{aligned}
\text{Invariants} \triangleq & \\
& \wedge \text{TotalVoted} = (\text{Cardinality}(\text{USERS}) * 3) - \text{SumVotesLeft}
\end{aligned}$$

$$\begin{aligned}
\text{WinnerIsFound} \triangleq & \\
& \Diamond(\text{winner} \in \text{MOVIES})
\end{aligned}$$

\ * Modification History
\ * Last modified *Thu Mar 25 10:45:43 CET 2021* by *davd*
\ * Created *Wed Mar 24 12:26:25 CET 2021* by *davd*