MODULE *Chameneos*

A specification of a 'concurrency game' requiring concurrent and symmetrical cooperation - https://*cedric.cnam.fr*/fichiers/*RC474.pdf*

EXTENDS *Integers*

RECURSIVE $Sum(\_, \_)$
$Sum(f, S) \triangleq$ IF $S = \{\}$ THEN 0
ELSE LET $x \triangleq$ CHOOSE $x \in S$: TRUE
IN $f[x] + Sum(f, S \setminus \{x\})$

$Color \triangleq \{$"blue", "red", "yellow"$\}$
$Faded \triangleq$ CHOOSE $c : c \notin Color$

$Complement(c1, c2) \triangleq$ IF $c1 = c2$
THEN $c1$
ELSE CHOOSE $cid \in Color \setminus \{c1, c2\}$: TRUE

N - number of total meeting after which *chameneoses* fade
M - number of *chameneoses*

CONSTANT N, M
ASSUME $N \in (Nat \setminus \{0\}) \land M \in (Nat \setminus \{0\})$

VARIABLE *chameneoses*, *meetingPlace*, *numMeetings*

$vars \triangleq \langle chameneoses, meetingPlace, numMeetings \rangle$

$ChameneosID \triangleq 1 \,..\, M$
$MeetingPlaceEmpty \triangleq$ CHOOSE $e : e \notin ChameneosID$

$TypeOK \triangleq$
For each *chameneoses*, remember its current color and how many meetings it has been in.
$\land\ chameneoses \in [ChameneosID \to (Color \cup \{Faded\}) \times (0 \,..\, N)]$
A *meetingPlace* (called Mall in the original paper) keeps track of the *chameneoses* creature that is currently waiting to meet another creature.
$\land\ meetingPlace \in ChameneosID \cup \{MeetingPlaceEmpty\}$

$Init \triangleq \land\ chameneoses \in [ChameneosID \to Color \times \{0\}]$
$\land\ meetingPlace = MeetingPlaceEmpty$
$\land\ numMeetings = 0$

$Meet(cid) \triangleq$ IF $meetingPlace = MeetingPlaceEmpty$
THEN IF $numMeetings < N$
chameneos enters empty meeting place
THEN $\land\ meetingPlace' = cid$
$\land$ UNCHANGED $\langle chameneoses, numMeetings \rangle$

chameneos takes on faded color

ELSE $\land\ chameneoses' = [chameneoses \text{ EXCEPT } ![cid] = \langle Faded,\ @[2]\rangle]$

$\land$ UNCHANGED $\langle meetingPlace,\ numMeetings\rangle$

meeting place is not empty - two *chameneoses* mutate

ELSE $\land\ meetingPlace \neq cid$

$\land\ meetingPlace' = MeetingPlaceEmpty$

$\land\ chameneoses' =$

LET $newColor \triangleq Complement(chameneoses[cid][1],$
$chameneoses[meetingPlace][1])$

IN $[chameneoses \text{ EXCEPT } ![cid] = \langle newColor,\ @[2]+1\rangle,$
$![meetingPlace] = \langle newColor,\ @[2]+1\rangle]$

$\land\ numMeetings' = numMeetings + 1$

Repeatedly try to enter meeting place for *chameneoses* that are not faded yet.
The system terminates once the color of all *chameneoses* is faded.

$Next \triangleq \land \exists\, c \in \{x \in ChameneosID : chameneoses[x][1] \neq Faded\} : Meet(c)$

$Spec \triangleq Init \land \Box[Next]_{vars}$

Upon termination, the sum of the (individual) meetings that all creates have
been in, is equal to $2 * N$. It is *not* guaranteed that all *chameneoses* have
been in a meeting with another *chameneoses*. See section A. Game termination
on page 5 of the original papaer).

$SumMet \triangleq numMeetings = N \Rightarrow$ LET $f[c \in ChameneosID] \triangleq chameneoses[c][2]$
IN $Sum(f,\ ChameneosID) = 2 * N$

THEOREM $Spec \Rightarrow \Box SumMet$
