```
#|

 Copyright (C) 1994 by Sakthi Subramanian.  All Rights Reserved.

 This script is hereby placed in the public domain, and therefore unlimited
 editing and redistribution is permitted.

 NO WARRANTY

 Sakthi Subramanian PROVIDES ABSOLUTELY NO WARRANTY.  THE EVENT
 SCRIPT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER
 EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, ANY IMPLIED
 WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.
 THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE SCRIPT IS
 WITH YOU.  SHOULD THE SCRIPT PROVE DEFECTIVE, YOU ASSUME THE COST OF
 ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

 IN NO EVENT WILL Sakthi Subramanian BE LIABLE TO YOU FOR ANY
 DAMAGES, ANY LOST PROFITS, LOST MONIES, OR OTHER SPECIAL, INCIDENTAL
 OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE
 THIS SCRIPT (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
 RENDERED INACCURATE OR LOSSES SUSTAINED BY THIRD PARTIES), EVEN IF
 YOU HAVE ADVISED US OF THE POSSIBILITY OF SUCH DAMAGES, OR FOR ANY
 CLAIM BY ANY OTHER PARTY.

|#
```

A Mechanically Checked Proof of the
Mutilated Checkerboard Theorem

Sakthi Subramanian

Here is a formalization of the nxn mutilated checkerboard problem using Bob Boyer's representation. I later learnt that Herb Simon uses a similar trick in discussing a solution to a "Cube-brick problem". Our formalization allows us to prove by induction that every sequence of domino-placements leads to a state in which the number of covered black and white squares are equal. However, in the mutilated board the number of black and white squares are not equal and hence it cannot be covered completely. An interesting feature is the definition of the color predicates in terms of the representation of the squares. White squares are those whose coordinates sum up to an even number and black squares are those whose coordinates sum upto an odd number. We prove the theorem for a nxn board using mathematical induction.

REPRESENTATION:

We label the squares by their coordinates with (0,0) being the upper left hand corner and (n,n) being the diagonally opposite square. Thus, there are n+1 rows and columns in the board. If n=7, we are talking about the usual 8x8 checkerboard with the rows and columns numbered from 0 through 7. More generally, when n is odd, the number of rows and columns is even and vice versa. We use [0..m, 0..n] as our notation for a board whose rows are numbered from 0 through m and whose columns are numbered from 0 through n.

EVENT: Start with the initial **nqthm** theory.

A square is represented by a pair $\mathrm{cons}\,(x,\ y)$ where x and y are its coordinates.

DEFINITION:
$\mathrm{squarep}\,(x) = (\mathrm{listp}\,(x) \land (\mathrm{car}\,(x) \in \mathbf{N}) \land (\mathrm{cdr}\,(x) \in \mathbf{N}))$

A domino is a pair of adjacent squares. Two squares are adjacent if one of their coordinates is the same and the other coordinate differs by one. Our predicate is true provided the sum of the coordinates of the first argument is less than that of the second argument. Thus, our definition is asymmetric but this will do: there is no loss of generality.

DEFINITION:
$\mathrm{adjp}\,(s1,\ s2)$
$= \quad (((\mathrm{car}\,(s1) = \mathrm{car}\,(s2)) \land ((1 + \mathrm{cdr}\,(s1)) = \mathrm{cdr}\,(s2)))$
$\qquad \lor \quad ((\mathrm{cdr}\,(s1) = \mathrm{cdr}\,(s2)) \land ((1 + \mathrm{car}\,(s1)) = \mathrm{car}\,(s2))))$

A square x in a [0..n,0..n] board is a pair of coordinates each less than or equal to n.

DEFINITION:
$\mathrm{squarenp}\,(x,\ n) = (\mathrm{squarep}\,(x) \land (\mathrm{car}\,(x) \le n) \land (\mathrm{cdr}\,(x) \le n))$

Dominoes are adjacent pairs of squares that fall within the board.

DEFINITION:
dominop $(x,\ n)$
$=$ (squarenp (car (x), n) $\wedge$ squarenp (cdr (x), n) $\wedge$ adjp (car (x), cdr (x)))

Square-listp recognizes a list of squares.

DEFINITION:
square-listp (x)
$=$ **if** $x \simeq$ **nil then t**
else squarep (car (x)) $\wedge$ square-listp (cdr (x)) **endif**

A board state is a list of covered squares.

DEFINITION: board-statep (x) = square-listp (x)

Having modeled all possible states of all possible checkerboards let us move on to actions. There is exactly one action: place a domino with the preconditions that the squares of the domino should not be already covered. We don't have to check if the dominoes fall within the board because the dominop definition takes care of it.

The state got as a result of placing a domino x on a board in state s is given by the following function. If the preconditions of the action are not satisfied then it returns a non-board-statep whose car is `'failed`. This "error" state is returned when other actions are performed on it. Thus, a legal domino placement action is one that results in a board state.

DEFINITION:
res-place $(x,\ s)$
$=$ **if** (car $(x) \in s$) $\vee$ (cdr $(x) \in s$)
then list (`'failed`, `'place`, x, s)
else cons (car (x), cons (cdr (x), s)) **endif**

DEFINITION:
result $(a,\ s)$
$=$ **if** car (s) = `'failed` **then** s
else res-place (cadr (a), s) **endif**

The state got from s as a result of executing a list of actions l is given by the following function.

DEFINITION:
resultlist $(l,\ s)$
$=$ **if** $l \simeq$ **nil then** s
else resultlist (cdr (l), result (car (l), s)) **endif**

A constructor for the action is given below.

DEFINITION: place (x) = list $($'place$, x)$

The following are predicates on domino-placement actions and sequences of them.

DEFINITION: placep (x, n) = dominop (cadr (x), n)

DEFINITION:
place-planp (x, n)
= **if** $x \simeq$ **nil** **then t**
else placep (car (x), n) $\wedge$ place-planp (cdr (x), n) **endif**

To express the theorem, we need a predicate on states in which all squares except the corner squares (0,0) and (n,n) are covered. We write functions to generate the set of all squares on the board and then delete the two corner squares from the set.

Make-row constructs the set of all squares in the row numbered m from columns 0 through n.

DEFINITION:
make-row (m, n)
= **if** $n \simeq 0$ **then** list (cons $(m, 0)$)
else append (make-row $(m, n-1)$, list (cons (m, n))) **endif**

Make-all-rows constructs the set of squares in rows 0 through m and columns 0 through n.

DEFINITION:
make-all-rows (m, n)
= **if** $m \simeq 0$ **then** make-row $(0, n)$
else append (make-all-rows $(m-1, n)$, make-row (m, n)) **endif**

The following function deletes the first occurrence of x in l.

DEFINITION:
delete (x, l)
= **if** $l \simeq$ **nil** **then** l
elseif x = car (l) **then** cdr (l)
else cons (car (l), delete $(x$, cdr (l))) **endif**

A mutilated [0..n,0..n] board includes all squares in rows 0 through n except the squares '(0 . 0) and cons (n, n).

DEFINITION:
mutilated-board (n)
= delete (cons (n, n), delete ('(0 . 0), make-all-rows (n, n)))

DEFINITION:
set-equal $(l1, l2)$
$=$ **if** $l1 \simeq$ **nil** **then** $l2 \simeq$ **nil**
else $(\text{car}(l1) \in l2)$
$\wedge$ set-equal (cdr $(l1)$, delete (car $(l1)$, $l2$)) **endif**

A state in which all squares of the mutilated [0..n,0..n] board are covered is given by the following predicate.

DEFINITION:
all-covered-except-cornerp (x, n) = set-equal $(x$, mutilated-board $(n))$

A white square is one whose coordinates add up to an even number. Otherwise it is a black square. Predicates on white and black squares are given below.

DEFINITION: oddp $(x) = ((x$ **mod** $2) = 1)$

DEFINITION: whitep $(x) = (((\text{car}(x) + \text{cdr}(x))$ **mod** $2) = 0)$

DEFINITION: blackp $(x) = (((\text{car}(x) + \text{cdr}(x))$ **mod** $2) = 1)$

The following functions compute the number of white and black squares in a state.

DEFINITION:
nwhite (x)
$=$ **if** $x \simeq$ **nil** **then** 0
elseif whitep (car (x)) **then** 1 + nwhite (cdr (x))
else nwhite (cdr (x)) **endif**

DEFINITION:
nblack (x)
$=$ **if** $x \simeq$ **nil** **then** 0
elseif blackp (car (x)) **then** 1 + nblack (cdr (x))
else nblack (cdr (x)) **endif**

THE PROOF.

What is the impossibility argument? In the desired state the number of covered white squares is not equal to number of covered black squares. Every placement operation covers exactly one white square and one black square. Thus, all states reachable starting with a state in which there are no dominoes on the board have an equal number of covered white and black squares. Ergo the desired state is unachievable.

First we show that make-all-rows (n, n) has an equal number of white and black squares when n is odd (theorem eq-bw-board1) and that the number of white squares = number of black squares + 1 when n is even (theorem white-one-plus-black2). The lemmas appearing before them were needed for the proofs.

THEOREM: add-before-sub1
$(x \geq 2) \rightarrow ((1 + (x - 2)) = ((1 + x) - 2))$

Odd numbers succeed even numbers.

THEOREM: odd-succeed-even1
$((n \ \mathbf{mod} \ 2) = 0) \rightarrow (((1 + n) \ \mathbf{mod} \ 2) = 1)$

THEOREM: move-1-out
$(m + (1 + n)) = (1 + (m + n))$

If a square is white then the next square is black.

THEOREM: black-follows-white1
$\text{whitep}\,(\text{cons}\,(m,\ n)) \rightarrow \text{blackp}\,(\text{cons}\,(m,\ 1 + n))$

Even numbers succeed odd numbers.

THEOREM: even-succeed-odd1
$((n \ \mathbf{mod} \ 2) = 1) \rightarrow (((1 + n) \ \mathbf{mod} \ 2) = 0)$

The square next to a black square is white.

THEOREM: white-follows-black1
$\text{blackp}\,(\text{cons}\,(m,\ n)) \rightarrow \text{whitep}\,(\text{cons}\,(m,\ 1 + n))$

THEOREM: odd-even1
$((n \ \mathbf{mod} \ 2) \neq 0) \rightarrow ((n \ \mathbf{mod} \ 2) = 1)$

The following is the base case of the lemma that the number of white squares equals the number of black squares in a row with an even number of squares.

THEOREM: expand-make-row1
$\text{make-row}\,(m,\ 1) = \text{list}\,(\text{cons}\,(m,\ 0),\ \text{cons}\,(m,\ 1))$

THEOREM: equal-bw-row-base0
$\text{nwhite}\,(\text{make-row}\,(m,\ 1)) = \text{nblack}\,(\text{make-row}\,(m,\ 1))$

THEOREM: t5
$((y \ \mathbf{mod} \ 2) \neq 1) \rightarrow (((1 + y) \ \mathbf{mod} \ 2) = 1)$

THEOREM: append-assoc
$\text{append}\,(\text{append}\,(l1,\ l2),\ l3) = \text{append}\,(l1,\ \text{append}\,(l2,\ l3))$

THEOREM: t6
$$(n \geq 2) \rightarrow (\text{make-row}\,(m,\ n) = \text{append}\,(\text{make-row}\,(m,\ n - 2),\ \text{list}\,(\text{cons}\,(m,\ n - 1),\ \text{cons}\,(m,\ n))))$$

The number of white/black squares obtained on appending two lists is the sum of the number of white/black squares of the individual lists.

THEOREM: nwhite-append1
nwhite (append ($l1$, $l2$)) = (nwhite ($l1$) + nwhite ($l2$))

THEOREM: nblack-append1
nblack (append ($l1$, $l2$)) = (nblack ($l1$) + nblack ($l2$))

In each row of a chess board [0..m,0..n] where n is odd, the number of black squares is equal to the number of white squares.

THEOREM: equal-bw-row1
oddp (n) $\rightarrow$ (nwhite (make-row (m, n)) = nblack (make-row (m, n)))

In an entire [0..m,0..n] board, n odd, there are an equal number of white and black squares.

THEOREM: eq-bw-board1
oddp (n) $\rightarrow$ (nwhite (make-all-rows (m, n)) = nblack (make-all-rows (m, n)))

Instantiating m as n in the above we get the following.

THEOREM: eq-bw-board
oddp (n) $\rightarrow$ (nwhite (make-all-rows (n, n)) = nblack (make-all-rows (n, n)))

Now we prove that the number of white squares = number of black squares + 1 in a board with an odd number of rows and columns (theorem white-one-plus-black2). The following are intermediate lemmas needed for the proof.

DEFINITION: evenp (x) = ((x **mod** 2) = 0)

THEOREM: zero-ident
(m + 0) = fix (m)

The number of white squares of an even numbered row with an odd number of columns is equal to the number of black squares + 1.

THEOREM: white-one-plus-black
(evenp (m) $\wedge$ evenp (n))
$\rightarrow$ (nwhite (make-row (m, n)) = (1 + nblack (make-row (m, n))))

THEOREM: black-one-plus-white
(oddp (m) $\wedge$ evenp (n))
$\rightarrow$ (nblack (make-row (m, n)) = (1 + nwhite (make-row (m, n))))

THEOREM: l13
make-all-rows (1, n) = append (make-row (0, n), make-row (1, n))

THEOREM: l14
$\text{evenp}(n) \rightarrow (\text{nwhite}(\text{make-all-rows}(\mathtt{1}, n)) = \text{nblack}(\text{make-all-rows}(\mathtt{1}, n)))$

THEOREM: base2
$$(\text{nwhite}(\text{make-row}(\mathtt{0}, n)) + \text{nwhite}(\text{make-row}(\mathtt{1}, n))) = (\text{nblack}(\text{make-row}(\mathtt{0}, n)) + \text{nblack}(\text{make-row}(\mathtt{1}, n)))$$

THEOREM: l15
$$(m > \mathtt{0}) \rightarrow ((\text{nwhite}(\text{make-row}(m - 1, n)) + \text{nwhite}(\text{make-row}(m, n))) = (\text{nblack}(\text{make-row}(m - 1, n)) + \text{nblack}(\text{make-row}(m, n))))$$

THEOREM: t61
$$(m \geq \mathtt{2}) \rightarrow (\text{make-all-rows}(m, n) = \text{append}(\text{make-all-rows}(m - \mathtt{2}, n), \text{append}(\text{make-row}(m - 1, n), \text{make-row}(m, n))))$$

THEOREM: l16
$$((x1 + y1 + z1) = (x1 + y2 + z2)) = ((y1 + z1) = (y2 + z2))$$

In a [0..m,0..n] checkerboard in which m is odd, the number of black squares is equal to the number of white squares.

THEOREM: eq-bw-board2
$\text{oddp}(m) \rightarrow (\text{nwhite}(\text{make-all-rows}(m, n)) = \text{nblack}(\text{make-all-rows}(m, n)))$

THEOREM: l17
$(\text{evenp}(m) \land (m > \mathtt{0})) \rightarrow (((m - 1) \ \mathbf{mod}\ \mathtt{2}) = \mathtt{1})$

The following function is introduced to force Nqthm to chose the appropriate induction scheme for theorem white-one-plus-black2. A hint to induct on m according to the recursive structure of f1 was given.

DEFINITION:
f1 (x)
= **if** $x \simeq \mathtt{0}$ **then t**
else f1 $(x - 1)$ **endif**

If the number of rows and columns of a checkerboard is odd then the number of white squares in the board is 1 greater than the number of black squares.

THEOREM: white-one-plus-black2
$$(\text{evenp}(m) \land \text{evenp}(n)) \rightarrow (\text{nwhite}(\text{make-all-rows}(m, n)) = (1 + \text{nblack}(\text{make-all-rows}(m, n))))$$

Having established that the number of white squares is equal to the number of black squares in a board with an even number of rows and columns and that the number of white squares is 1 greater than the number of black squares in a board with an odd number of rows and columns, we would like to show that in a mutilated board the number of white squares is less than the number of black squares. This is proved as theorem mut3 below. The lemmas preceding it are intermediate lemmas needed for the proof.

THEOREM: t7
$(n \in \mathbf{N}) \rightarrow (\text{cons}\,(m,\ n) \in \text{make-row}\,(m,\ n))$

THEOREM: t8
$(x \in \mathit{l2}) \rightarrow (x \in \text{append}\,(\mathit{l1},\ \mathit{l2}))$

THEOREM: m1
$((n \in \mathbf{N}) \land (m \in \mathbf{N})) \rightarrow (\text{cons}\,(m,\ n) \in \text{make-all-rows}\,(m,\ n))$

If we delete a white square from a list then the number of white squares in the new list is less than that in the old list.

THEOREM: white-delete
$((x \in l) \land \text{whitep}\,(x)) \rightarrow (\text{nwhite}\,(\text{delete}\,(x,\ l)) < \text{nwhite}\,(l))$

The (n,n) square is white.

THEOREM: white1
$\text{whitep}\,(\text{cons}\,(n,\ n))$

Deleting a white square does not change the number of black squares.

THEOREM: black-same1
$\text{whitep}\,(x) \rightarrow (\text{nblack}\,(\text{delete}\,(x,\ l)) = \text{nblack}\,(l))$

Deleting `'(0 . 0)` and cons $(n,\ n)$ does not change the number of black squares in a list since both of them are white.

THEOREM: black-del1
$\text{nblack}\,(\text{delete}\,(\text{cons}\,(n,\ n),\ \text{delete}\,(\texttt{'(0 . 0)},\ x))) = \text{nblack}\,(x)$

THEOREM: mut-l2
$\text{oddp}\,(n) \rightarrow (n \neq \texttt{0})$

THEOREM: mem-del1
$((x \in l) \land (x \neq y)) \rightarrow (x \in \text{delete}\,(y,\ l))$

THEOREM: m2
$\texttt{'(0 . 0)} \in \text{make-all-rows}\,(m,\ n)$

THEOREM: mut-l1
$\text{oddp}(n) \rightarrow (\text{cons}(n,\ n) \in \text{delete}(\texttt{'(0 . 0)},\ \text{make-all-rows}(n,\ n)))$

In a mutilated board with an even number of rows and columns the number of white squares is less than the number of black squares.

THEOREM: mut1
$(\text{oddp}(n) \wedge (x = \text{mutilated-board}(n))) \rightarrow (\text{nwhite}(x) < \text{nblack}(x))$

In a mutilated board with an odd number of rows and columns, the number of white squares is less than the number of black squares.

THEOREM: mut2
$(\text{evenp}(n) \wedge (x = \text{mutilated-board}(n)) \wedge (n > \texttt{0}))$
$\rightarrow \quad (\text{nwhite}(x) < \text{nblack}(x))$

In any mutilated board the number of white squares is less than the number of black squares.

THEOREM: mut3
$((x = \text{mutilated-board}(n)) \wedge (n > \texttt{0})) \rightarrow (\text{nwhite}(x) < \text{nblack}(x))$

The following shows what a mutilated board looks like for n = 7.

THEOREM: mutboard7
mutilated-board (7)

```
=   '((0 . 1)
      (0 . 2)
      (0 . 3)
      (0 . 4)
      (0 . 5)
      (0 . 6)
      (0 . 7)
      (1 . 0)
      (1 . 1)
      (1 . 2)
      (1 . 3)
      (1 . 4)
      (1 . 5)
      (1 . 6)
      (1 . 7)
      (2 . 0)
      (2 . 1)
      (2 . 2)
      (2 . 3)
```

```
(2 . 4)
(2 . 5)
(2 . 6)
(2 . 7)
(3 . 0)
(3 . 1)
(3 . 2)
(3 . 3)
(3 . 4)
(3 . 5)
(3 . 6)
(3 . 7)
(4 . 0)
(4 . 1)
(4 . 2)
(4 . 3)
(4 . 4)
(4 . 5)
(4 . 6)
(4 . 7)
(5 . 0)
(5 . 1)
(5 . 2)
(5 . 3)
(5 . 4)
(5 . 5)
(5 . 6)
(5 . 7)
(6 . 0)
(6 . 1)
(6 . 2)
(6 . 3)
(6 . 4)
(6 . 5)
(6 . 6)
(6 . 7)
(7 . 0)
(7 . 1)
(7 . 2)
(7 . 3)
(7 . 4)
(7 . 5)
(7 . 6))
```

We will now use mut3 to show that in all states satisfying all-covered-except-cornerp the number of white squares is less than the number of black squares. This is proved as theorem unequal3 below. We need some more lemmas for the proof because all-covered-except-cornerp checks for set equality by deleting squares in two given lists.

Deleting a non-white element from a list of white squares does not alter the number of white squares.

THEOREM: white-same1
$(\neg\ \text{whitep}\,(x)) \rightarrow (\text{nwhite}\,(\text{delete}\,(x,\ l)) = \text{nwhite}\,(l))$

THEOREM: white-del1
$((x \in l) \wedge \text{whitep}\,(x)) \rightarrow (\text{nwhite}\,(l) = (1 + \text{nwhite}\,(\text{delete}\,(x,\ l))))$

If two sets of squares are equal then the number of white/black squares in them are equal.

THEOREM: nwhite-eq1
$\text{set-equal}\,(l1,\ l2) \rightarrow (\text{nwhite}\,(l1) = \text{nwhite}\,(l2))$

THEOREM: black-same2
$(\neg\ \text{blackp}\,(x)) \rightarrow (\text{nblack}\,(\text{delete}\,(x,\ l)) = \text{nblack}\,(l))$

THEOREM: black-del2
$((x \in l) \wedge \text{blackp}\,(x)) \rightarrow (\text{nblack}\,(l) = (1 + \text{nblack}\,(\text{delete}\,(x,\ l))))$

THEOREM: nblack-eq1
$\text{set-equal}\,(l1,\ l2) \rightarrow (\text{nblack}\,(l1) = \text{nblack}\,(l2))$

If all the squares except the corner ones are covered in a board state then the number of white squares covered is less than the number of covered black squares.

THEOREM: unequal1
$(\text{evenp}\,(n)$
$\wedge\quad (n > 0)$
$\wedge\quad \text{board-statep}\,(x)$
$\wedge\quad \text{all-covered-except-cornerp}\,(x,\ n))$
$\rightarrow\quad (\text{nwhite}\,(x) < \text{nblack}\,(x))$

THEOREM: unequal2
$(\text{oddp}\,(n) \wedge \text{board-statep}\,(x) \wedge \text{all-covered-except-cornerp}\,(x,\ n))$
$\rightarrow\quad (\text{nwhite}\,(x) < \text{nblack}\,(x))$

THEOREM: unequal3
$((n > 0) \wedge \text{board-statep}\,(x) \wedge \text{all-covered-except-cornerp}\,(x,\ n))$
$\rightarrow\quad (\text{nwhite}\,(x) < \text{nblack}\,(x))$

Now we want to show that the number of covered white squares is equal to the number of covered black squares in every state that arises starting with an initial state when there are no dominoes on the board. This is proved as theorem bcequal1. This in turn uses theorem bw-equal2 which says that the number of covered black and white squares remain equal when a domino is placed on a board legally.

First we need some more number theory: If the sum of two numbers is even then adding one to one of the numbers will make the sum odd and vice versa.

THEOREM: t10
$((1 + w) + z) = (1 + (w + z))$

THEOREM: t11
$(x + 1 + y) = (1 + (x + y))$

A domino covers exactly one white square and one black square.

THEOREM: domino-white1
$\text{dominop}\,(x,\ n)$
$\rightarrow \quad (\text{nwhite}\,(\text{cons}\,(\text{car}\,(x),\ \text{cons}\,(\text{cdr}\,(x),\ y))) = (1 + \text{nwhite}\,(y)))$

THEOREM: domino-black1
$\text{dominop}\,(x,\ n)$
$\rightarrow \quad (\text{nblack}\,(\text{cons}\,(\text{car}\,(x),\ \text{cons}\,(\text{cdr}\,(x),\ y))) = (1 + \text{nblack}\,(y)))$

If a domino is placed on a board then the new board state will have the two squares under the domino covered.

THEOREM: place-dom1
$(\text{board-statep}\,(s) \wedge \text{placep}\,(a,\ n) \wedge \text{board-statep}\,(\text{result}\,(a,\ s)))$
$\rightarrow \quad (\text{result}\,(a,\ s) = \text{cons}\,(\text{caadr}\,(a),\ \text{cons}\,(\text{cdadr}\,(a),\ s)))$

If the number of covered black and white squares in a state are equal then they remain the same after a legal domino placement.

THEOREM: bw-equal2
$(\text{board-statep}\,(s)$
$\wedge \quad (\text{nwhite}\,(s) = \text{nblack}\,(s))$
$\wedge \quad \text{placep}\,(a,\ n)$
$\wedge \quad \text{board-statep}\,(\text{result}\,(a,\ s)))$
$\rightarrow \quad (\text{nwhite}\,(\text{result}\,(a,\ s)) = \text{nblack}\,(\text{result}\,(a,\ s)))$

The following lemmas are need to show that if the number of covered white and black squares are equal in a state then they will remain equal after every legal sequence of domino-placement actions.

First we prove that executing an action sequence in an error state results in the error state.

THEOREM: failed-state1
$(\text{car}(s) = \texttt{'failed}) \rightarrow (\text{car}(\text{resultlist}(p, s)) = \texttt{'failed})$

THEOREM: res2
$(\text{board-statep}(s) \land \text{placep}(a, n) \land (\neg\ \text{board-statep}(\text{result}(a, s))))$
$\rightarrow \quad (\text{car}(\text{result}(a, s)) = \texttt{'failed})$

THEOREM: s1
$(\text{car}(s) = \texttt{'failed}) \rightarrow (\neg\ \text{board-statep}(s))$

THEOREM: res3
$(\text{board-statep}(s)$
$\land \quad \text{place-planp}(p, n)$
$\land \quad \text{listp}(p)$
$\land \quad \text{board-statep}(\text{resultlist}(p, s)))$
$\rightarrow \quad \text{board-statep}(\text{result}(\text{car}(p), s))$

The following function is used to force the prover to choose the correct induction scheme for the theorem bcequal1.

DEFINITION:
$\text{foo}(s, p)$
$= \quad \textbf{if } p \simeq \textbf{nil} \textbf{ then nil}$
$\quad \textbf{else } \text{foo}(\text{result}(\text{car}(p), s), \text{cdr}(p)) \textbf{ endif}$

If the number of covered white and black squares is equal in a state and a sequence of legal domino placement operations is executed then the number of covered white and black squares would be equal in the resulting state.

THEOREM: bcequal1
$(\text{board-statep}(s)$
$\land \quad (\text{nwhite}(s) = \text{nblack}(s))$
$\land \quad \text{place-planp}(p, n)$
$\land \quad \text{board-statep}(\text{resultlist}(p, s)))$
$\rightarrow \quad (\text{nwhite}(\text{resultlist}(p, s)) = \text{nblack}(\text{resultlist}(p, s)))$

The final theorem: Starting with an initial state s of an [0..n,0..n] board, n greater than 0, in which there are no covered squares there is no sequence of actions that will result in a state in which all the squares except (0,0) and (n,n) are covered.

THEOREM: tough-nut
$(\text{place-planp}(p, n)$
$\land \quad (n > 0)$
$\land \quad (s1 = \text{resultlist}(p, \textbf{nil}))$
$\land \quad \text{board-statep}(s1))$
$\rightarrow \quad (\neg\ \text{all-covered-except-cornerp}(s1, n))$

Index