```
#|
From kaufmann Fri Sep  6 12:11:08 1991
To: windley@cs.uidaho.edu
Subject: example

Hi, Phil --

I went ahead and did that example.  It took me just under an hour at
the terminal; I'd already written out a page of definitions and
lemmas, as indicated below (I think I showed them to you).  The final
theorem, main-theorem, says that there are arbitrarily large n such
that running the interpreter for n steps, from the initial state,
results in a return to the initial state.  But the content of that
theorem is really in main-lemma, which has no quantifiers and is where
the proof effort lies.

Often in such efforts I'll go back and get rid of the INSTRUCTIONS
hints, which are printed out by the system when a proof using my
interactive ``proof-checker'' is completed.  By ``get rid of'' I mean
that I'll try to prove suitable rewrite rules so that I can produce an
input file without INSTRUCTIONS hints, which can go through the
unadorned Boyer-Moore prover.  I haven't done that here, simply
because I can't see that it's worth the time (hard to tell how long it
would take).  So, I suspect that the ``proof'' I finally got is quite
similar to one you would obtain, except that the system helped me a
lot with the arithmetic (even though I didn't load in any libraries),
inductions were easy to hint, and I could do lots of simplifications
at the subterm level.

Anyhow, below is the input file, which takes about 53 seconds of
real time to replay through pc-nqthm on a Sun 3/60.
-- Matt
|#
```

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

```
;; I had preliminary definitions of step, interp, and interp-prop written
;; out on paper when starting this, along with statements of the lemmas
;; interp-plus and main-lemma.  I'll time the rest....
```

DEFINITION:
step $(s,\ d0)$

$=$ **let** *state* **be** car (s),
d **be** cdr (s)
in
case on *state*:
case $=$ `0`
then cons $(\texttt{1},\ d0)$
case $=$ `1`
then if `0` $< d$ **then** cons $(\texttt{2},\ d)$
else cons $(\texttt{3},\ d)$ **endif**
case $=$ `2`
then cons $(\texttt{1},\ d - 1)$
otherwise cons $(\texttt{0},\ \texttt{0})$ **endcase endlet**

DEFINITION:
interp $(s,\ d0,\ n)$
$=$ **if** $n \simeq$ `0` **then** s
else interp $(\text{step}\,(s,\ d0),\ d0,\ n - 1)$ **endif**

THEOREM: interp-plus
interp $(s,\ d0,\ m + n) =$ interp $(\text{interp}\,(s,\ d0,\ m),\ d0,\ n)$

THEOREM: plus-assoc
$((x + y) + z) = (x + (y + z))$

THEOREM: interp-from-2
$(w \not\simeq \texttt{0}) \rightarrow (\text{interp}\,(\text{cons}\,(\texttt{2},\ d),\ d0,\ w) = \text{interp}\,(\text{cons}\,(\texttt{1},\ d - 1),\ d0,\ w - 1))$

THEOREM: numberp-cdr-interp
$((d0 \in \mathbf{N}) \wedge (\text{cdr}\,(\textit{cons-state-d}) \in \mathbf{N}))$
$\rightarrow$ $(\text{cdr}\,(\text{interp}\,(\textit{cons-state-d},\ d0,\ w)) \in \mathbf{N})$

THEOREM: through-the-loop
$((d \in \mathbf{N}) \wedge (d0 \in \mathbf{N}))$
$\rightarrow$ $(\text{interp}\,(\text{cons}\,(\texttt{1},\ d),\ d0,\ d + d + w) = \text{interp}\,(\texttt{'(1 . 0)},\ d0,\ w))$

THEOREM: main-lemma
$(d0 \in \mathbf{N})$
$\rightarrow$ $(\text{interp}\,(\texttt{'(0 . 0)},\ d0,\ k * (\texttt{3} + d0 + d0)) = \texttt{'(0 . 0)})$

```
#| The following generates the four forms following it:
(defn-sk+ interp-prop (d0)
  (forall x (exists n (and (lessp x n)
   (equal (interp (cons 0 0) d0 n)
  (cons 0 0))))))
|#
```

DEFINITION:
interp-prop $(d0)$
$\leftrightarrow \quad \forall\, x\, \exists\, n\, ((x < n) \wedge (\text{interp}\,(\text{cons}\,(\texttt{0}, \texttt{0}),\ d0,\ n) = \text{cons}\,(\texttt{0}, \texttt{0})))$

EVENT: Disable interp-prop.

THEOREM: interp-prop-suff
$((\text{x}\,(d0) < n) \wedge (\text{interp}\,(\texttt{'(0 . 0)},\ d0,\ n) = \texttt{'(0 . 0)}))$
$\rightarrow \quad \text{interp-prop}\,(d0)$

THEOREM: interp-prop-necc
$(\neg\, ((x < \text{n}\,(d0,\ x)) \wedge (\text{interp}\,(\texttt{'(0 . 0)},\ d0,\ \text{n}\,(d0,\ x)) = \texttt{'(0 . 0)})))$
$\rightarrow \quad (\neg\, \text{interp-prop}\,(d0))$

THEOREM: main-theorem-hack-lemma
$(x < ((1 + x) * (\texttt{3} + w))) = \mathbf{t}$

THEOREM: main-theorem
$(d0 \in \mathbf{N}) \rightarrow \text{interp-prop}\,(d0)$

Index