185 (sort test) Write a program to assign a binary variable to indicate whether a given list is sorted.

After trying the question, scroll down to the solution.

§ Let L be the list (a constant), and let b be the binary state variable whose final value will indicate whether L is sorted. Let n be a natural state variable.

For $0 \le n \le \#L$, define *sorted n* to mean that $L\,[n;..\#L]$ is sorted.

$$\textit{sorted } n \;=\; n{=}\#L \vee \forall i{:}\; n{+}1,..\#L{\cdot}\; L(i{-}1) \le L\, i$$

Now refine.

$$b' = \textit{sorted}\ 0 \;\wedge\; t' \le t + \#L \;\Longleftarrow\; n{:=}\,0.\;\; n{\le}\#L \;\Rightarrow\; b' = \textit{sorted } n \;\wedge\; t' \le t + \#L - n$$

$$n{\le}\#L \;\Rightarrow\; b' = \textit{sorted } n \;\wedge\; t' \le t + \#L - n \;\Longleftarrow$$
$$\quad \textbf{if } \#L{-}n \le 1 \textbf{ then } b{:=}\,\top$$
$$\quad \textbf{else if } L(n{-}1) > L\, n \textbf{ then } b{:=}\,\bot$$
$$\quad\quad \textbf{else } n{:=}\,n{+}1.\;\; t{:=}\,t{+}1.\;\; n{\le}\#L \;\Rightarrow\; b' = \textit{sorted } n \;\wedge\; t' \le t + \#L - n \textbf{ fi fi}$$

Proof of the first refinement, starting with its right side.

$n{:=}\,0.\;\; n{\le}\#L \;\Rightarrow\; b' = \textit{sorted } n \;\wedge\; t' \le t + \#L - n$ — substitution law

$0{\le}\#L \;\Rightarrow\; b' = \textit{sorted}\ 0 \;\wedge\; t' \le t + \#L - 0$ — simplify

$=\quad b' = \textit{sorted}\ 0 \;\wedge\; t' \le t + \#L$

Proof of the last refinement in three cases. First case:

$(n{\le}\#L \;\Rightarrow\; b' = \textit{sorted } n \;\wedge\; t' \le t + \#L - n \;\Longleftarrow\; \#L{-}n \le 1 \;\wedge\; (b{:=}\,\top))$ — portation

$=\quad \#L{-}n \le 1 \;\wedge\; (b{:=}\,\top) \;\wedge\; n{\le}\#L \;\Rightarrow\; b' = \textit{sorted } n \;\wedge\; t' \le t + \#L - n$ — assignment

$=\quad \#L{-}n \le 1 \;\wedge\; b' \;\wedge\; n'{=}n \;\wedge\; t'{=}t \;\wedge\; n{\le}\#L \;\Rightarrow\; b' = \textit{sorted } n \;\wedge\; t' \le t + \#L - n$ — simplify

$=\quad (n{=}\#L \vee n{=}\#L{-}1) \;\wedge\; b' \;\wedge\; n'{=}n \;\wedge\; t'{=}t \;\Rightarrow\; b' = \textit{sorted } n \;\wedge\; t' \le t + \#L - n$ — distribute and antidistribute

$=\quad (n{=}\#L \;\wedge\; b' \;\wedge\; n'{=}n \;\wedge\; t'{=}t \;\Rightarrow\; b' = \textit{sorted } n \;\wedge\; t' \le t + \#L - n)$
$\quad \wedge\; (n{=}\#L{-}1 \;\wedge\; b' \;\wedge\; n'{=}n \;\wedge\; t'{=}t \;\Rightarrow\; b' = \textit{sorted } n \;\wedge\; t' \le t + \#L - n)$ — top line: $n{=}\#L \;\wedge\; t'{=}t \;\Rightarrow\; t' \le t + \#L - n$; bottom line: $n{=}\#L{-}1 \;\wedge\; t'{=}t \;\Rightarrow\; t' \le t + \#L - n$

$=\quad (n{=}\#L \;\wedge\; b' \;\wedge\; n'{=}n \;\wedge\; t'{=}t \;\Rightarrow\; b' = \textit{sorted } n)$
$\quad \wedge\; (n{=}\#L{-}1 \;\wedge\; b' \;\wedge\; n'{=}n \;\wedge\; t'{=}t \;\Rightarrow\; b' = \textit{sorted } n)$ — expand *sorted*

$=\quad (n{=}\#L \;\wedge\; b' \;\wedge\; n'{=}n \;\wedge\; t'{=}t \;\Rightarrow\; b' = (n{=}\#L \vee \forall i{:}\; n{+}1,..\#L{\cdot}\; L(i{-}1){\le}L\, i))$
$\quad \wedge\; (n{=}\#L{-}1 \;\wedge\; b' \;\wedge\; n'{=}n \;\wedge\; t'{=}t \;\Rightarrow\; b' = (n{=}\#L \vee \forall i{:}\; n{+}1,..\#L{\cdot}\; L(i{-}1){\le}L i))$ — top line: use $n{=}\#L$ context; bottom line: use context $n{=}\#L{-}1$ in antecedent makes $\forall$ domain empty

$=\quad (n{=}\#L \;\wedge\; b' \;\wedge\; n'{=}n \;\wedge\; t'{=}t \;\Rightarrow\; b' = (\top \vee \forall i{:}\; n{+}1,..\#L{\cdot}\; L(i{-}1){\le}L\, i))$
$\quad \wedge\; (n{=}\#L{-}1 \;\wedge\; b' \;\wedge\; n'{=}n \;\wedge\; t'{=}t \;\Rightarrow\; b' = (n{=}\#L \vee \top)$

$=\quad (n{=}\#L \;\wedge\; b' \;\wedge\; n'{=}n \;\wedge\; t'{=}t \;\Rightarrow\; b')$
$\quad \wedge\; (n{=}\#L{-}1 \;\wedge\; b' \;\wedge\; n'{=}n \;\wedge\; t'{=}t \;\Rightarrow\; b')$ — specialize, reflexive, idempotent

$=\quad \top$

Proof of the last refinement middle case:

$($ $n{\le}\#L \;\Rightarrow\; b' = \textit{sorted } n \;\wedge\; t' \le t + \#L - n$

$\Longleftarrow\quad \#L{-}n > 1 \;\wedge\; L(n{-}1) > L\, n \;\wedge\; (b{:=}\,\bot))$ — portation, assignment, simplify

$=\quad \#L{-}n > 1 \;\wedge\; L(n{-}1) > L\, n \;\wedge\; \neg b' \;\wedge\; n'{=}n \;\wedge\; t'{=}t \;\Rightarrow\; b' = \textit{sorted } n \;\wedge\; t' \le t + \#L - n$ — $\#L{-}n > 1 \;\wedge\; t'{=}t \;\Rightarrow\; t' \le t + \#L - n$

$=\quad \#L{-}n > 1 \;\wedge\; L(n{-}1) > L\, n \;\wedge\; \neg b' \;\wedge\; n'{=}n \;\wedge\; t'{=}t \;\Rightarrow\; b' = \textit{sorted } n$ — expand *sorted*

$=\quad \#L{-}n > 1 \;\wedge\; L(n{-}1) > L\, n \;\wedge\; \neg b' \;\wedge\; n'{=}n \;\wedge\; t'{=}t$
$\quad \Rightarrow\; b' = (n{=}\#L \vee \forall i{:}\; n{+}1,..\#L{\cdot}\; L(i{-}1) \le L\, i)$

$\#L–n > 1$ makes $n=\#L$ false, and
$\#L–n > 1$ makes the $\forall$ domain nonempty, and
$L(n–1) > L\,n$ makes $\forall i\colon n+1,..\#L\cdot\ L(i–1) \le L\,i$ false

$=$ $\#L–n > 1 \ \wedge\ L(n–1) > Ln \ \wedge\ \neg b' \ \wedge\ n'=n \ \wedge\ t'=t$
$\Rightarrow\ b' = (\bot \vee \bot)$ idempotent, specialize
$=$ $\top$

Proof of the last refinement last case:

$($ $n \le \#L \ \Rightarrow\ b' = sorted\ n \ \wedge\ t' \le t + \#L - n$
$\Leftarrow$ $\#L–n > 1 \ \wedge\ L(n–1) \le L\,n$
$\wedge\ (n:= n+1.\ t:= t+1.\ n \le \#L \ \Rightarrow\ b' = sorted\ n \ \wedge\ t' \le t + \#L - n))$ substitution
$=$ $($ $n \le \#L \ \Rightarrow\ b' = sorted\ n \ \wedge\ t' \le t + \#L - n$
$\Leftarrow$ $\#L–n > 1 \ \wedge\ L(n–1) \le L\,n$
$\wedge\ (n+1 \le \#L \ \Rightarrow\ b' = sorted\ (n+1) \ \wedge\ t' \le t + 1 + \#L - (n+1)))$ simplify
$=$ $($ $n \le \#L \ \Rightarrow\ b' = sorted\ n \ \wedge\ t' \le t + \#L - n$
$\Leftarrow$ $\#L–n > 1 \ \wedge\ L(n–1) \le L\,n$
$\wedge\ (n+1 \le \#L \ \Rightarrow\ b' = sorted\ (n+1) \ \wedge\ t' \le t + \#L - n))$ discharge
$=$ $($ $n \le \#L \ \Rightarrow\ b' = sorted\ n \ \wedge\ t' \le t + \#L - n$
$\Leftarrow$ $\#L–n > 1 \ \wedge\ L(n–1) \le L\,n$
$\wedge\ b' = sorted\ (n+1) \ \wedge\ t' \le t + \#L - n)$ portation
$=$ $\#L–n > 1 \ \wedge\ L(n–1) \le L\,n \ \wedge\ b' = sorted\ (n+1) \ \wedge\ t' \le t + \#L - n \ \wedge\ n \le \#L$
$\Rightarrow\ b' = sorted\ n \ \wedge\ t' \le t + \#L - n$ simplify and specialize
$\Leftarrow$ $\#L–n > 1 \ \wedge\ L(n–1) \le L\,n \ \wedge\ b' = sorted\ (n+1) \ \Rightarrow\ b' = sorted\ n$
expand *sorted* , quantifier law, reflexive
$\Leftarrow$ $\top$