467 A binary tree can be stored as a list of nodes in breadth order. Traditionally, the root is at index 1 , the node at index n has its left child at index $2{\times}n$ and its right child at index $2{\times}n{+}1$. Suppose the user's variable is $x{:}\ X$, and the implementer's variables are $s{:}\ [{*}X]$ and $p{:}\ nat{+}1$, and the operations are

$$
\begin{array}{lll}
goHome & = & p := 1 \\
goLeft & = & p := 2{\times}p \\
goRight & = & p := 2{\times}p + 1 \\
goUp & = & p := div\ p\ 2 \\
put & = & s := p{\rightarrow}x \mid s \\
get & = & x := s\ p
\end{array}
$$

Now suppose we decide to move the entire list down one index so that we do not waste index 0 . The root is at index 0 , its children are at indexes 1 and 2 , and so on. Develop the necessary data transform, and use it to transform the operations.

After trying the question, scroll down to the solution.

§   The new implementer's variables are r: [*X] and q: *nat* . The transform is
$$r = s[1;..\#s] \;\wedge\; p = q+1$$
For each of the transformations, it will be easy enough to eliminate the three variables p , s' , and p' by one-point. The trick to eliminate s is explained after the transformations.

Transform *goHome* :
$$\forall s, p \cdot \qquad r = s[1;..\#s] \wedge p = q+1$$
$$\Rightarrow \quad \exists s', p' \cdot r' = s'[1;..\#s'] \wedge p' = q'+1 \wedge x'=x \wedge s'=s \wedge p'=1$$
$$= \quad x'=x \wedge r'=r \;\wedge\; 1 = q'+1$$
$$= \quad q := 0$$

Transform *goLeft* :
$$\forall s, p \cdot \qquad r = s[1;..\#s] \wedge p = q+1$$
$$\Rightarrow \quad \exists s', p' \cdot r' = s'[1;..\#s'] \wedge p' = q'+1 \wedge x'=x \wedge s'=s \wedge p'=2\times p$$
$$= \quad x'=x \wedge r'=r \;\wedge\; 2\times(q+1) = q'+1$$
$$= \quad q := 2\times q + 1$$

Transform *goRight* :
$$\forall s, p \cdot \qquad r = s[1;..\#s] \wedge p = q+1$$
$$\Rightarrow \quad \exists s', p' \cdot r' = s'[1;..\#s'] \wedge p' = q'+1 \wedge x'=x \wedge s'=s \wedge p'=2\times p+1$$
$$= \quad x'=x \wedge r'=r \;\wedge\; 2\times(q+1) + 1 = q'+1$$
$$= \quad q := 2\times q + 2$$

Transform *goUp* :
$$\forall s, p \cdot \qquad r = s[1;..\#s] \wedge p = q+1$$
$$\Rightarrow \quad \exists s', p' \cdot r' = s'[1;..\#s'] \wedge p' = q'+1 \wedge x'=x \wedge s'=s \wedge p' = div\ p\ 2$$
$$= \quad x'=x \wedge r'=r \;\wedge\; div\ (q+1)\ 2 = q'+1$$
$$= \quad q := div\ (q+1)\ 2 - 1$$
$$= \quad q := div\ (q-1)\ 2$$

Transform *put* :
$$\forall s, p \cdot \qquad r = s[1;..\#s] \wedge p = q+1$$
$$\Rightarrow \quad \exists s', p' \cdot r' = s'[1;..\#s'] \wedge p' = q'+1 \wedge x'=x \wedge s'=p\rightarrow x \mid s \wedge p'=p$$
$$= \quad x'=x \wedge r' = q\rightarrow x \mid r \;\wedge\; q'+1=q+1$$
$$= \quad r := q\rightarrow x \mid r$$

Transform *get* :
$$\forall s, p \cdot \qquad r = s[1;..\#s] \wedge p = q+1$$
$$\Rightarrow \quad \exists s', p' \cdot r' = s'[1;..\#s'] \wedge p' = q'+1 \wedge x'=s\ p \wedge s'=s \wedge p'=p$$
$$= \quad x'=r\ q \wedge r'=r \;\wedge\; q'+1=q+1$$
$$= \quad x := r\ q$$

To transform *put* we start with

$$\forall s, p \cdot \qquad r = s[1;..\#s] \wedge p = q+1$$
$$\Rightarrow \quad \exists s', p' \cdot r' = s'[1;..\#s'] \wedge p' = q'+1 \wedge x'=x \wedge s'=p\rightarrow x \mid s \wedge p'=p$$

The three variables p , s' , and p' are easy to eliminate by one-point. We get

$\quad = \quad \forall s \cdot\ r = s[1;..\#s] \quad \Rightarrow \quad r' = (q{+}1{\rightarrow}x \mid s)[1;..\#(q{+}1{\rightarrow}x \mid s)] \wedge q{+}1 = q'{+}1 \wedge x'{=}x$

The problem is to get rid of s because we don't have $s{=}$something . We have
$$r = s[1;..\#s]$$
From this we see that $\#r = \#s{-}1$ and $s = [i];;r$ for some unknown item i . I'll use that to eliminate s .

$\quad = \quad r' = (q{+}1{\rightarrow}x \mid [i];;r)[1;..\#(q{+}1{\rightarrow}x \mid [i];;r)] \wedge q{+}1 = q'{+}1 \wedge x'{=}x$

We can simplify $\#(q{+}1{\rightarrow}x \mid [i];;r)$ ro $\#r{+}1$ and simplify $q{+}1 = q'{+}1$ to $q'{=}q$.

$\quad = \quad r' = (q{+}1{\rightarrow}x \mid [i];;r)[1;..\ \#r{+}1] \wedge q'{=}q \wedge x'{=}x$

Now we need to simplify $(q{+}1{\rightarrow}x \mid [i];;r)[1;..\ \#r{+}1]$. We have a list $(q{+}1{\rightarrow}x \mid [i];;r)$ of length $\#r{+}1$, and in this list at index $q{+}1$ the item is x . Now we index with the list $[1;..\ \#r{+}1]$, which shifts all the indexes down 1 . So now at index q the item is x .

$\quad = \quad x'{=}x \wedge r' = q{\rightarrow}x \mid r \wedge q'{=}q$
$\quad = \quad r := q{\rightarrow}x \mid r$

I wish I could see a nice series of formal steps.