Assignment 3

Ruofei HUANG(z5141448) Anqi ZHU(z5141541)

May 12, 2018

1 Task 1

According to the specified problem statement in the assignment, we could describe the syntactic data type $Dict$ as below. The encapsulated state is a dictionary word set W.

$$Dict = (W = \phi,$$
$$\left(\begin{array}{l} \textbf{proc } addword^{Dict}(\textbf{value } w) \cdot b, W : [\ \text{TRUE}, b = b_0 \wedge W = W_0 \cup \{w\}] \\ \textbf{func } checkword^{Dict}(\textbf{value } w) : \mathbb{B} \cdot \textbf{var } b \cdot\ b, W : [\ \text{TRUE}, b = (w \in W_0)];\ \textbf{return } b \\ \textbf{proc } delword^{Dict}(\textbf{value } w) \cdot b, W : [w \in W, b = b_0 \wedge W = W_0 \backslash \{w\}] \end{array} \right))$$

2 Task 2

Now we would like to refine $Dict$ to a second data type $DictA$ where we replace W with a trie t, the corresponding trie domain $\mathbf{dom}t$ and a counter i that holds the index of the next free cell in the domain t array. We also borrow the definition of $\mathcal{T}$ in the problem statement. It represents the set of all tries according to the domain. We shall use this definition later in our refinement.
/*probably need to define a data invariant here*/
This suggests we should first build up a inductively defined predicate to ensure the provable relations between $DictA$ and $Dict$.

$$\begin{array}{l} \text{r} = \ ((\epsilon \mapsto 1) \in t \vee i = 0 \Leftrightarrow W = \phi) \wedge \\ ((\epsilon \mapsto 0) \in t \wedge i > 0 \Leftrightarrow W = \exists S(W = \\ \left(\begin{array}{l} (domt[i-1] \cup S \wedge r[^S/_W][^{i-1}/_i] \wedge t[i-1] = domt[i-1] \mapsto 1) \wedge \\ (S \wedge r[^S/_W][^{i-1}/_i] \wedge t[i-1] = domt[i-1] \mapsto 0) \end{array} \right))) \end{array}$$

which we can translate into a function from concrete to abstract values:

$\cup f(t, i-1)$if $(\epsilon \mapsto 0) \in t \wedge i > 0 \wedge t[i-1] = domt[i-1] \mapsto 1$
$f(t, i-1)$otherwise

With that in mind we propose the initialisation predicate $init^{DictA} = (i = 0)$ and operations given as follows.

continue