Assignment 1

z5141448 Ruofei HUANG

April 5, 2018

Definations

Two Dimensional Arrays Expression

To describe variants of two-dimensional arrays we write $(b : k \mapsto j \mapsto x)$ instead of $(b : k \mapsto (b[k] : j \mapsto x))$. We use this new notation to state an instance of the array-assignment axiom we saw already

$$\left\{\phi[^{(b:k\mapsto x)}/_{b}]\right\} b[k] := x \left\{\phi\right\}$$

for two-dimensional arrays:

$$\left\{\phi[^{(b:k\mapsto j\mapsto x)}/_{b}]\right\} b[k][j] := x \left\{\phi\right\}$$

String Length

A string $S \in \mathit{Letter}^*$ which is an array of letters[1].Also, string will be terminate by the null character which is a convention by the C programming language and we will follow this convention in this proof. We write $|S|$ for the number of letters in the string. Formally, we define these two nothion inductively by

$$|S\ell| = |S| + \begin{cases} 1 & \text{if } \ell \neq' \backslash 0' \\ 0 & \text{if } \ell =' \backslash 0' \end{cases}$$

Also, by the convention of C we has this definition for $S \in \mathit{string}$.

$$S[|S|] =' \backslash 0' \wedge \forall 0 \leq i < |S| \,(S[i] \neq' \backslash 0')$$

[1]The letter here is a legal charater encode with ASCII, UTF-8 or other charater encoding standard.

String Equals

To describe two string a, b $(a, b \in String)$ are equals we write $a = b$ when:

$$a = b \iff |a| = |b| \wedge \forall j \in 0..\,|a|\,(a[j] = b[j])$$

Similarly, we write:

$$a \neq b \iff \neg(a = b)$$

String Assign

To assign a string to another string array, we will denote as

$$a := b$$

instead of a long programme of our toy language:

$\{a, b \in String\}$
$\{I[^0/_i]\}$
$i := 0;$
$\{I\}$
while $i \leq |b|$ **do**
    $\{I \wedge i \leq |b|\}$
    $\{I[^{i+1}/_i][^{a:i \mapsto b[i]}/_a]\}$
    $a[i] := b[i];$
    $\{I[^{i+1}/_i]\}$
    $i := i + 1;$
    $\{I\}$
od;
$\{I \wedge i > |b|\}$
$\{a, b \in String \wedge a = b\}$

when our invariant is

$$I = a, b \in String \wedge 0 \leq i \leq (|b| + 1) \wedge \forall k \in 0..(i-1)\,(a[k] = b[k])$$

Here are the proofs of the implications:

First Implication for String assign: $a, b \in String \Rightarrow I[^0/_i]$

$a, b \in String$
$\Rightarrow$ $\langle$using $|b| \in \mathbb{N}$ and realising that the last conjunct is vacuously true$\rangle$
$a, b \in String \wedge 0 \leq 0 \leq (|b| + 1) \wedge \forall k \in 0..(0-1)\,(a[k] = b[k])$
$\Leftrightarrow$ $\langle$definition of I and substitution$\rangle$
$I[^0/_i]$

Second Implication $I \wedge i \leq |b| \Rightarrow I[^{i+1}/_i][^{a:i\mapsto b[i]}/_a]$

We first look at the LHS:

$$
\begin{aligned}
& I \wedge i \leq |b| \\
\Leftrightarrow \quad & \langle \text{Substitue I } \rangle \\
& a, b \in String \wedge 0 \leq i \leq (|b| + 1) \wedge \forall k \in 0..(i-1)\,(a[k] = b[k]) \wedge i \leq |b| \\
\Leftrightarrow \quad & \langle \text{Conjunct } i \leq (|b| + 1) \text{ and } i \leq |b| \rangle \\
& a, b \in String \wedge 0 \leq i \leq |b| \wedge \forall k \in 0..(i-1)\,(a[k] = b[k])
\end{aligned}
$$

We then expand RHS:

$$
\begin{aligned}
& I[^{i+1}/_i][^{a:i\mapsto b[i]}/_a] \\
\Leftrightarrow \quad & \langle \text{substitute } i = i+1 \text{ and } a[i] = b[i] \text{ by definition} \rangle \\
& a, b \in String \wedge 0 \leq i+1 \leq (|b| + 1) \wedge \forall k \in 0..((i+1)-1)\,(a[k] = b[k]) \wedge a[i] := b[i]
\end{aligned}
$$

We then have a clear imply

$$
\begin{aligned}
& a, b \in String \wedge 0 \leq i \leq |b| \wedge \forall k \in 0..(i-1)\,(a[k] = b[k]) \\
\Rightarrow \quad & \langle i \leq |b| \Rightarrow i+1 \leq |b| + 1 \text{ and } a[i] := b[i] \ \rangle \\
& a, b \in String \wedge 0 \leq i+1 \leq (|b| + 1) \wedge \forall k \in 0..((i+1)-1)\,(a[k] = b[k]) \wedge a[i] := b[i]
\end{aligned}
$$

Third Implication $I \wedge i > |b| \Rightarrow a, b \in String \wedge a = b$

$$
\begin{aligned}
& I \wedge i > |b| \\
\Leftrightarrow \quad & \langle \text{substitution of I } \rangle \\
& a, b \in String \wedge 0 \leq i \leq (|b| + 1) \wedge \forall k \in 0..(i-1)\,(a[k] := b[k]) \wedge i > |b| \\
\Leftrightarrow \quad & \langle\ i > |b| \text{and } i \leq (|b| + 1) \text{ with some calculation} \rangle \\
& a, b \in String \wedge \forall k \in 0..\,|b|\,(a[k] := b[k]) \\
\Rightarrow \quad & \langle \text{Definition of two string equal} \rangle \\
& a, b \in String \wedge a = b
\end{aligned}
$$

String Compare

Missing part

1 Task 1

Since we have define some manipulation of String, we can see a string as a whole. So the input is an array of String. Hence we can define our precondiction as:

$$a \in String^* \wedge |a| = n$$

As the post condition as:

$$a, b \in String^* \wedge |b| \leq |a| \wedge \forall i \in 1..(|a|-1), a[i] \neq a[i+1], a[i-1] \neq a[i]$$
$$(\exists j \in 1..(|b|-1)\,(a[i-1] = b[j-1], a[i] = b[j], a[i+1] = b[j+1], i \geq j))$$
$$\wedge \forall j \in 1..length(b)\,(b[j] \neq b[j-1] \wedge b[j-1] \neq b[j+1])$$

The postcondition mainly say all the adjust line is not the same string in array a , then the b array will contain those identical adjust three line and all the adjust line in the array b is not equal. Hence the post condition is not friendly there is a refinement for this postcondition that would be use in programme this uniq.

1.1 Refinement of the postcondition

By having this observation(proof)

$$b \in String^* \wedge j \in 0..(|b|-1) \forall b\,([)\,j] \neq b[j+1]$$
$$\Leftrightarrow \qquad \langle \text{sss} \rangle$$

we can refine the post condition to this easier one.

$$LOL$$

2 Task 2

3 Task 3

4 Task 4