Assignment 3: Derivation

Maxwell Bo 43926871

May 17, 2017

1. (a) n is a **value** parameter. m is a **result** parameter.

   (b) $inv \triangleq lrun(A, n_0, m)$

   (c) Let

$$\begin{aligned} pre(A, n) &\triangleq lrun(A, n, n+1) \\ inv(A, n, m) &\triangleq lrun(A, n_0, m) \\ post(A, n, m) &\triangleq mrun(A, n_0, m) \end{aligned}$$

   by 1.(b), and the specification of the procedure. *inv*, *pre* and *post* implicitly capture variables (A, n, m) as parameters from the frame, for syntactic convience.
   s.t.

$$\begin{aligned} & n, m : [pre,\ post] \\ \sqsubseteq\ & \{\text{Composition: middle predicate is } inv\} \\ & n, m : [pre,\ inv];\quad n, m : [inv,\ post] \\ \sqsubseteq\ & \{\text{Assignment: } pre \Rrightarrow inv[m \backslash n+1]\} \\ & m := n+1;\quad n, m : [inv,\ post] \end{aligned}$$

   $\because$

$$\begin{aligned} inv[m \backslash n+1] &\equiv lrun(A, n_0, m)[m \backslash n+1] \\ &\equiv lrun(A, n_0, n+1) \end{aligned}$$

   $\therefore$

$$lrun(A, n, n+1) \Rrightarrow lrun(A, n_0, n+1)$$

   Let

$$guard \triangleq (m < A.\text{len} \wedge A_{n_0} = A_m)$$

   s.t.

$$\begin{aligned} \sqsubseteq\ & \{\text{Strengthen post: } inv \wedge \neg guard \Rrightarrow post\} \\ & m := n+1;\quad n, m : [inv,\ inv \wedge \neg guard] \end{aligned}$$

   $\because$

$$\begin{aligned} & inv \wedge \neg guard \Rrightarrow post \\ \equiv\ & \{\text{Expansion of definitions}\} \\ & lrun(A, n_0, m) \wedge \neg(m < A.\text{len} \wedge A_{n_0} = A_m) \Rrightarrow mrun(A, n_0, m) \\ \equiv\ & \{\text{Expansion of functions}\} \\ & lrun(A, n_0, m) \wedge \neg(m < A.\text{len} \wedge A_{n_0} = A_m) \Rrightarrow lrun(A, n_0, m) \wedge (m < A.\text{len} \Rightarrow A_{n_0} \neq A_m) \\ \equiv\ & \{\text{De Morgan's law - negation of conjunction}\} \end{aligned}$$

$$
\begin{array}{ll}
& \mathit{lrun}(A, n_0, m) \wedge (\neg(m < A.\text{len}) \vee \neg(A_{n_0} = A_m)) \;\Rrightarrow\; \mathit{lrun}(A, n_0, m) \wedge (m < A.\text{len} \Rightarrow A_{n_0} \neq A_m) \\
\equiv & \{P \Rightarrow Q \;\equiv\; \neg P \vee Q\} \\
& \mathit{lrun}(A, n_0, m) \wedge (\neg(m < A.\text{len}) \vee \neg(A_{n_0} = A_m)) \;\Rrightarrow\; \mathit{lrun}(A, n_0, m) \wedge (\neg(m < A.\text{len}) \vee (A_{n_0} \neq A_m)) \\
\equiv & \{\} \\
& \text{true}
\end{array}
$$

$$
\begin{array}{ll}
\sqsubseteq & \{\text{Repetition}\} \\
& m := n + 1; \\
& \textbf{do } (m < A.\text{len} \wedge A_{n_0} = A_m) \rightarrow \\
& \quad n, m : [\mathit{inv} \wedge \mathit{guard},\ \mathit{inv} \wedge (0 \leqslant V < V_0)] \\
& \textbf{od}
\end{array}
$$

where

$$V \;\triangleq\; A.\text{len} \;-\; m$$

$$
\begin{array}{ll}
\sqsubseteq & \{\text{Assignment: } \mathit{inv} \wedge \neg \mathit{guard} \;\Rrightarrow\; (\mathit{inv} \wedge (0 \leqslant V < V_0))[m \backslash m + 1]\} \\
& m := n + 1; \\
& \textbf{do } (m < A.\text{len} \wedge A_{n_0} = A_m) \rightarrow \\
& \quad m := m + 1 \\
& \textbf{od}
\end{array}
$$

$\ddots$

$$
\begin{array}{lll}
(\mathit{inv} \wedge (0 \leqslant V < V_0))[m \backslash m + 1] & \equiv & (\mathit{lrun}(A, n_0, m) \wedge (0 \leqslant (A.\text{len} - m) < (???)))[m \backslash m + 1] \\
& \equiv & \mathit{lrun}(A, n_0, m + 1) \wedge (0 \leqslant (A.\text{len} - (m + 1)) < (???))
\end{array}
$$

TODO

2.

$$
\begin{array}{rcl}
\mathit{pre}(A) & \triangleq & A.\text{len} > 0 \\
\mathit{post}(A, \ell, h) & \triangleq & \mathit{mrun}(A, \ell, h) \wedge (\forall\, p, q \;\cdot\; \mathit{mrun}(A, p, q) \Rightarrow (h - \ell) \leqslant (q - p))
\end{array}
$$

by 2. *pre* and *post* implicitly capture variables (A, ℓ, h) as parameters from the frame.

$$
\begin{array}{ll}
& \ell, h : [\mathit{pre},\ \mathit{post}] \\
\sqsubseteq & \{\text{Composition: middle predicate is } \mathit{inv}\} \\
& \ell, h : [\mathit{pre},\ \mathit{inv}];\ \ \ell, h : [\mathit{inv},\ \mathit{post}]
\end{array}
$$

where

$$\mathit{inv} \;\triangleq\; \text{TODO}$$

where *inv* implicitly captures varibles (TODO as parameters from the frame.