8.6 Case Study: Find Positive Element More Quickly

Review 7.4 (P261)

- Consider an integer array a and a constant $N \geq 1$. The task is to write a program *FIND* that finds the smallest index $k \in \{1, . . ., N\}$ with $a[k] > 0$ if such an element of a exists; otherwise the dummy value $k = N + 1$ should be returned.
- Formally, the program *FIND* should satisfy the input/output specification

  $\{true\}$

  FIND

  $\{1 \leq k \leq N + 1 \wedge \forall(1 \leq l < k) : a[l] \leq 0 \wedge (k \leq N \rightarrow a[k] > 0)\}$

  in the sense of total correctness. Clearly, we require $a \notin change(FIND)$.

Question 8.6 (P291)

- In case study 8.6, we consider an improved program *FINDPOS* for the same problem.
- Thus it should satisfy the correctness formula

$$\{true\}$$

$$FINDPOS \tag{8.17}$$

$$\{1 \leq k \leq N + 1 \wedge \forall(0 < l < k) : a[l] \leq 0 \wedge (k \leq N \rightarrow a[k] > 0)\}$$

in the sense of total correctness, where $a \notin change(FINDPOS)$.

Question 8.6 (P291)

- Just as in *FIND*, the program *FINDPOS* consists of two components S_1 and S_2 activated in parallel.
- S_1 searches for an odd index k of a positive element and S_2 searches for an even one.
- **What is new** is that now S_1 should stop searching once S_2 has found a positive element and vice versa for S_2. Thus some communication should take place between S_1 and S_2.

Question 8.6 (P291)

- Thus the program *FINDPOS* is of the form

$$FINDPOS \equiv i := 1;\ j := 2;\ oddtop := N + 1;\ eventop := N + 1;$$

$$[S_1 \parallel S_2];$$

$$k := min(oddtop, eventop)$$

oddtop and eventop are shared variables

- where

$S_1 \equiv$ **while** $i < min(oddtop, eventop)$ **do**

if $a[i] > 0$ **then** $oddtop := i$

else $i := i + 2$ **fi**

od

$S_2 \equiv$ **while** $j < min(oddtop, eventop)$ **do**

if $a[j] > 0$ **then** $eventop := j$

else $j := j + 2$ **fi**

od

Proof Outlines —— Find Invariants and Bound Functions

- Let p_1, p_2 and t_1, t_2 be the invariants and bound functions introduced in Section 7.4; that is,
- $p_1 \equiv \quad 1 \le oddtop \le N + 1 \wedge odd(i) \wedge 1 \le i \le oddtop + 1$

  $\wedge \quad \forall l : (odd(l) \wedge 1 \le l < i \rightarrow a[l] \le 0)$

  $\wedge \quad (oddtop \le N \rightarrow a[oddtop] > 0)$
- $\boldsymbol{t_1 \equiv oddtop + 1 - i}$
- $p_2 \equiv \quad 2 \le eventop \le N + 1 \wedge even(j) \wedge j \le eventop + 1$

  $\wedge \quad \forall l : (even(l) \wedge 1 \le l < j \rightarrow a[l] \le 0)$

  $\wedge \quad (eventop \le N \rightarrow a[eventop] > 0)$
- $\boldsymbol{t_2 \equiv eventop + 1 - j}$

$\{true\}$
$FINDPOS$
$\{1 \le k \le N + 1 \wedge \forall(0 < l < k) : a[l] \le 0 \wedge (k \le N \rightarrow a[k] > 0)\}$

Proof Outlines —— For Total Correctness (For S_1)

$\{\mathbf{inv} : p_1\}\{\mathbf{bd} : t_1\}$

$\mathbf{while}\ i < min(oddtop, eventop)\ \mathbf{do}$

$\quad \{p_1 \wedge \boxed{i < oddtop}\}$

$\quad \mathbf{if}\ a[i] > 0\ \ \mathbf{then}\ \{p_1 \wedge i < oddtop \wedge a[i] > 0\}$

$\qquad \{1 \leq i \leq N + 1 \wedge odd(i) \wedge 1 \leq i \leq i + 1$

$\qquad\quad \wedge\ \ \forall l : (odd(l) \wedge 1 \leq l < i \rightarrow a[l] \leq 0)$

$\qquad\quad \wedge\ \ (i \leq N \rightarrow a[i] > 0)\}$

$\qquad oddtop := i$

$\qquad \{p_1\}$

$\quad \mathbf{else}\ \ \{p_1 \wedge i < oddtop \wedge a[i] \leq 0\}$

$\qquad i := i + 2$

....

$\qquad \{p_1\}$

$\quad \mathbf{fi}$

$\quad \{p_1\}$

$\mathbf{od}$

$\{p_1 \wedge i \geq min(oddtop, eventop)\}$

i < min(oddtop, eventop) → i < oddtop

RULE 6: CONSEQUENCE

$$\frac{p \rightarrow p1,\ \{p1\}\ S\ \{q1\},\ q1 \rightarrow q}{\{p\}\ S\ \{q\}}$$

AXIOM 2: ASSIGNMENT

{p[u := t]} u := t {p}

$$\begin{aligned} p_1 \equiv\ & 1 \leq oddtop \leq N + 1 \wedge odd(i) \wedge 1 \leq i \leq oddtop + 1 \\ \wedge\ & \forall l : (odd(l) \wedge 1 \leq l < i \rightarrow a[l] \leq 0) \\ \wedge\ & (oddtop \leq N \rightarrow a[oddtop] > 0) \\ t_1 \equiv\ & oddtop + 1 - i \end{aligned}$$

Interference Freedom

- To apply the parallelism with shared variables rule 27 for the parallel composition of S_1 and S_2, we must show interference freedom of the two proof outlines.

$\{\textbf{inv}: p_1\}\{\textbf{bd}: t_1\}$
$\textbf{while}\ i < min(oddtop, eventop)\ \textbf{do}$
$\quad \{p_1 \wedge i < oddtop\}$
$\quad \textbf{if}\ a[i] > 0\ \textbf{then}\ \{p_1 \wedge i < oddtop \wedge a[i] > 0\}$
$\qquad \boxed{oddtop := i}$
$\quad \textbf{else}\ \{p_1 \wedge i < oddtop \wedge a[i] \le 0\}$
$\qquad \boxed{i := i + 2}$
$\quad \textbf{fi}$
$\textbf{od}$
$\{p_1 \wedge i \ge min(oddtop, eventop)\}$

$\{\textbf{inv}: p_2\}\{\textbf{bd}: t_2\}$
$\textbf{while}\ j < min(oddtop, eventop)\ \textbf{do}$
$\quad \{p_2 \wedge j < eventop\}$
$\quad \textbf{if}\ a[j] > 0\ \textbf{then}\ \{p_2 \wedge j < eventop \wedge a[j] > 0\}$
$\qquad \boxed{eventop := j}$
$\quad \textbf{else}\ \{p_2 \wedge j < eventop \wedge a[j] \le 0\}$
$\qquad \boxed{j := j + 2}$
$\quad \textbf{fi}$
$\textbf{od}$
$\{p_2 \wedge j \ge min(oddtop, eventop)\}$

- This amounts to checking 24 correctness formulas! Fortunately, ***22 of them*** are trivially satisfied because the variable changed by the assignment does not appear in the assertion or bound function under consideration.

Interference Freedom

$) \textbf{ then } \{p_2 \wedge j < eventop \wedge a[j] > 0\}$

$\boxed{eventop := j}$

$\wedge\ j < eventop \wedge a[j] < 0\}$

od

$\{p_1 \wedge i \geq min(oddtop, eventop)\}$

- $r = \{p_1 \wedge i \geq min(oddtop, eventop)\}$
- $R = eventop := j$
- $pre(R) = pre(eventop := j) = \{p_2 \wedge j < eventop \wedge a[j] > 0\}$

pre(R) → j < eventop

$\{p_1 \wedge i \geq min(oddtop, eventop) \wedge pre(eventop := j)\}$

$\{p_1 \wedge i \geq min(oddtop, eventop) \wedge \boxed{j < eventop}\}$

$\{p_1 \wedge i \geq min(oddtop, j)\}$

$eventop := j$

$\{p_1 \wedge i \geq min(oddtop, eventop)\}$

RULE 6: CONSEQUENCE

$$\frac{p \rightarrow p1,\ \{p1\}\ S\ \{q1\},\ q1 \rightarrow q}{\{p\}\ S\ \{q\}}$$

AXIOM 2: ASSIGNMENT

{p[u := t]} u := t {p}

{r ∧ pre(R)} R {r}

Satisfied!

Apply Rule 27

RULE 27: PARALLELISM WITH SHARED VARIABLES

$$\frac{\text{The standard proof outlines } \{p_i\}\ S_i^*\ \{q_i\},\ i \in \{1, \ldots, n\}, \text{ are interference free}}{\{\bigwedge_{i=1}^{n} p_i\}\ [S_1 \parallel \ldots \parallel S_n]\ \{\bigwedge_{i=1}^{n} q_i\}}$$

- An application of the parallelism with shared variables rule 27 now yields

$$\{p_1 \wedge p_2\}$$
$$[S_1 \parallel S_2]$$
$$\{p_1 \wedge p_2 \wedge i \geq min(oddtop, eventop) \wedge j \geq min(oddtop, eventop)\}$$

- By the assignment axiom and the consequence rule,

$$\{true\}$$
$$i := 1;\ j := 2;\ oddtop := N + 1;\ eventop := N + 1;$$
$$[S_1 \parallel S_2]$$
$$\{\ 1 \leq min(oddtop, eventop) \leq N + 1$$
$$\wedge\ \forall(0 < l < min(oddtop, eventop)) : a[l] \leq 0$$
$$\wedge\ (min(oddtop, eventop) \leq N \rightarrow a[min(oddtop, eventop)] > 0)\}$$
$$k := min(oddtop, eventop)$$
$$\{1 \leq k \leq N + 1 \wedge \forall(0 < l < k) : a[l] \leq 0 \wedge (k \leq N \rightarrow a[k] > 0)\}$$

$$\{true\}$$
$$FINDPOS$$
$$\{1 \leq k \leq N + 1 \wedge \forall(0 < l < k) : a[l] \leq 0 \wedge (k \leq N \rightarrow a[k] > 0)\}$$

Satisfied!

Thanks!