Chapter 6: Searching

A Problem

Problem: search for the maximal natural number i for which $i^2 \leq N$.

$$
\begin{aligned}
&\| [\\
&\mathbf{con}\ N; \{N \geq 0\} \\
&S \\
&\{x = (\mathbf{max}\ i : 0 \leq i \wedge i^2 \leq N : i)\} \\
&\,]\|.
\end{aligned}
$$

Note the post-condition may also be formulated as

$$\{x = (\mathbf{min}\ i : \underline{0 \leq i} \wedge \underline{(i+1)^2 > N} : i)\}$$

Searching for i meeting a certain condition ...

Linear Searching

Consider the following programming problem.

$$\|[$$
$$\mathbf{var}\ x : int;$$
$$\{(\exists i : 0 \leq i : b.i)\}$$
$$Linear\ Search$$
$$\{x = (\mathbf{min}\ i : 0 \leq i \wedge b.i : i)\}$$
$$]\|.$$

What is a possible invariant?

The post-condition can be rewritten as

$$R: \ 0 \leq x \wedge b.x \wedge (\forall i : 0 \leq i < x : \neg b.i)$$

So a possible invatiant is obtained by taking a conjunct:

$$P: \ 0 \leq x \wedge (\forall i : 0 \leq i < x : \neg b.i)$$

which is initialized by $x := 0$.

Investigation of $x := x + 1$ leads to

$$
\begin{array}{rl}
& P(x := x + 1) \\
\equiv & \{ \text{ definition of } P \} \\
& 0 \leq x + 1 \land (\forall i : 0 \leq i < x + 1 : \neg b.i) \\
\Leftarrow & \{ \text{ heading for } P \} \\
& 0 \leq x \land (\forall i : 0 \leq i < x + 1 : \neg b.i) \\
\equiv & \{ \text{ split off } i = x \} \\
& 0 \leq x \land (\forall i : 0 \leq i < x : \neg b.i) \land \neg b.x \\
\equiv & \{ \text{ definition of } P \} \\
& P \land \neg b.x
\end{array}
$$

The final program for linear search:

$$\begin{aligned}
&\|[\\
&\quad \textbf{var } x : int; \\
&\quad \{(\exists i : 0 \leq i : b.i)\} \\
&\quad x := 0; \\
&\quad \textbf{do } \neg b.x \rightarrow x := x + 1 \textbf{ od} \\
&\quad \{x = (\textbf{min } i : 0 \leq i \wedge b.i : i)\} \\
&\|].
\end{aligned}$$

Does this program terminate? Why?

Bounded Linear Search

The specification of the problem:

$$|[$$

$$\mathbf{con}\ N : int\ \{N \geq 0\}; b : \mathbf{array}\ [0..N)\ \mathbf{of}\ \mathrm{bool};$$

$$\mathbf{var}\ x : int;$$

bounded linear search

$$\{x = (\mathbf{max}\ i : 0 \leq i \leq N \wedge (\forall j : 0 \leq j < i : \neg b.j) : i)\}$$

$$]|$$

Could we use the invariant

$$0 \leq x \leq N \wedge (\forall i : 0 \leq i < x : \neg b.i)$$

to obtain the following program?

$$x := 0$$
$$\textbf{do } \neg b.x \wedge x \neq N \rightarrow x := x + 1 \textbf{ od}$$

No, since N does not belong to the domain of b and $x = N$ is not excluded by the invariant.

If we could define $b.N$ as true, the post-condition would be written as

$$R: \ 0 \leq x \leq N \wedge (\forall i : 0 \leq i < x : \neg b.i) \wedge b.x$$

Our idea is to define as invariant:

$$P_0: \ 0 \leq x \leq N \wedge (\forall i : 0 \leq i < x : \neg b.i) \wedge b.y$$
$$P_1: \ x \leq y \leq N$$

Then

$$P_0 \wedge P_1 \wedge x \neq y \rightarrow 0 \leq x < N,$$

so $b.x$ may occur in the statement of the repetition.

The final program:

$$
\begin{array}{l}
|[\\
\quad \textbf{var } y : int; \\
\quad x, y := 0, N; \\
\quad \textbf{do } x \neq y \rightarrow \\
\qquad \textbf{if } \neg b.x \rightarrow x := x + 1 \\
\qquad [] \; b.x \rightarrow y := x \\
\qquad \textbf{fi} \\
\quad \textbf{od} \\
]|
\end{array}
$$

Note that

$$
P_0 \wedge P_1 \wedge x \neq y \wedge \neg b.x \Rightarrow (P_0 \wedge P_1)(x := x + 1)
$$
$$
P_0 \wedge P_1 \wedge x \neq y \wedge b.x \Rightarrow (P_0 \wedge P_1)(y := x)
$$

Binary Search

Consider the problem:

$$\|[$$

$$\textbf{con } N, A : \textit{int } \{N \geq 1\}; \ \ f : \textbf{array } [0..N] \textbf{ of } \textit{int } \{f.0 \leq A < f.N\};$$

$$\textbf{var } x : \textit{int};$$

$$\textit{binary search}$$

$$\{f.x \leq A < f.(x+1)\}$$

$$\]|$$

Could we do better than linear search?

From the post-condition:

$$R: \quad f.x \leq A < f.(x+1)$$

we generalize $x+1$ to y, and define as invariants:

$$P_0: \quad f.x \leq A < f.y$$
$$P_1: \quad 0 \leq x < y \leq N$$

which is established by $x, y := 0, N$. And we may choose guard as

$$x + 1 \neq y.$$

What is a suitable bound function?

A straightforward bound function is $y - x$. To decrease it, we may choose a h such that $x < h < y$, and both

$$x := h$$

and

$$y := h$$

will decrease $y - x$.

How to keep the invariants?

We investigate the effects of $x := h$ and $y := h$ on the invariants.

$$P_0(x := h)$$
$$\equiv \quad \{ \text{ substitution } \}$$
$$f.h \leq A < f.y$$
$$\Leftarrow \quad \{ \text{ definition of } P_0 \}$$
$$P_0 \wedge f.h \leq A$$

$$P_0(y := h)$$
$$\equiv \quad \{ \text{ substitution } \}$$
$$f.x \leq A < f.h$$
$$\Leftarrow \quad \{ \text{ definition of } P_0 \}$$
$$P_0 \wedge A < f.h$$

This leads to

$$\{f.0 \le A < f.N\}$$

$$|[$$

var $y : intl$

$x, y := 0, N;$

$\{invarinat:\ P_0 \wedge P_1,\ bound:\ y - x\}$

do $x + 1 \neq y \rightarrow$

$\quad |[$

$\quad$ **var** $h : int;$

$\quad \underline{establish\ x < h < y};$

$\quad$ **if** $f.h \le A \rightarrow x := h$

$\quad\ [] \ A \le f.h \rightarrow y := h$

$\quad$ **fi**

$\quad]|$

od

$]|$

How to establish the underlined property?

So we obtain

$$x, y := 0, N;$$
$$\mathbf{do}\ x + 1 \neq y \rightarrow$$
$$\quad |[$$
$$\quad\quad \mathbf{var}\ h : int;$$
$$\quad\quad h = (x + y)\ \mathbf{div}\ 2;$$
$$\quad\quad \mathbf{if}\ f.h \leq A \rightarrow x := h$$
$$\quad\quad [ ]\ A < f.h \rightarrow y := h$$
$$\quad\quad \mathbf{fi}$$
$$\quad]|$$
$$\mathbf{od}$$

Two remarks:

- From the fact that $0 < h < N$, we know that $f.0$ and $f.N$ are not inspected.

- Precondition is only used for the initialization of x and y.

Square Root Problem

Consider writing a program for

$$|[$$
$$\mathbf{con}\ N : int\ \{N \geq 0\};$$
$$\mathbf{var}\ x : int;$$
$$square\ root$$
$$\{x^2 \leq N \land (x+1)^2 > N\}$$

whose time complexity is $\mathcal{O}(\log N)$.

Idea: Let $f.x = x * x$, which satisfies $f.x \leq f.(x+1)$ and $0 \leq N \leq f.N$. Applying the binary search yields:

$$
\begin{aligned}
&|[\\
&\quad \textbf{var } y : int; \\
&\quad x, y := 0, N + 1; \\
&\quad \textbf{do } x + 1 \neq y \rightarrow \\
&\quad\quad\quad |[\\
&\quad\quad\quad\quad \textbf{var } h : int; \\
&\quad\quad\quad\quad h = (x + y) \textbf{ div } 2; \\
&\quad\quad\quad\quad \textbf{if } h * h \leq N \rightarrow x := h \\
&\quad\quad\quad\quad\ []\ N < h * h \rightarrow y := h \\
&\quad\quad\quad\quad \textbf{fi} \\
&\quad\quad\quad]| \\
&\quad \textbf{od}
\end{aligned}
$$

The Binary Search Problem

Derive a program for

$$
\begin{aligned}
&|[\\
&\quad \textbf{con } N, A : int \; \{N \geq 1\}; \;\; f : \textbf{array } [0..N] \textbf{ of } int \; \{f.0 \leq A < f.N\}; \\
&\quad \{(\forall i, j : 0 \leq i \leq j < N : f.i \leq f.j)\} \\
&\quad \textbf{var } r : bool; \\
&\quad S \\
&\quad \{r \equiv (\exists i : 0 \leq i < N : f.i = A)\} \\
&]|
\end{aligned}
$$

Hint: Consider the post-condition as

$$
R : \; 0 \leq x < N \wedge (f.x \leq A < f.(x+1))
$$

while virtually assuming $f.N = \infty$.

Program S for "the binary search":

$$x, y := 0, N;$$
$$\textbf{do } x + 1 \neq y \rightarrow$$
$$\quad |[$$
$$\quad\quad \textbf{var } h : int;$$
$$\quad\quad h = (x + y) \textbf{ div } 2;$$
$$\quad\quad \textbf{if } f.h \leq A \rightarrow x := h$$
$$\quad\quad [] \ A \leq f.h \rightarrow y := h$$
$$\quad\quad \textbf{fi}$$
$$\quad]|$$
$$\textbf{od};$$
$$\underline{r := f.x = A}$$

Exercises

[Problem 7-1] Derive a program for the following specification.

$$|[$$

$$\textbf{con } N : int \ \{N \geq 0\};$$

$$\textbf{var } r : bool;$$

$$S$$

$$\{r \equiv (\exists p : \ p \geq 0 : N = p^3)\}$$

$$]|$$

[Problem 7-2] Derive for given N, $N \geq 0$, a program for the computation of the smallest integer x that satisfies $x^3 - 6x^2 + 9x \geq N$.

[Problem 7-3] Derive a program for the following specification.

$$\|[$$

$$\textbf{con } N : int \; \{N \geq 1\}; \;\; A, B : \textbf{array } [0..N] \textbf{ of } int;$$

$$\{A.0 \leq B.0 \wedge A.N \geq B.N\}$$

$$\textbf{var } r : int;$$

$$S$$

$$\{0 \leq r < N \wedge A.r \leq B.r \wedge A.(r+1) \geq B.(r+1)\}$$

$$\]|$$