Chapter 9. Segment Problems

Longest Segment Problems

$|[$

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

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

$maxseg$

$\{r = (\mathbf{max}\ p, q : 0 \leq p \leq q \leq N \wedge \mathcal{A}.p.q :\ q - p)\}$

$]|$

Examples

- all elements are constant:

$$\mathcal{A}.p.q = (\forall i, j : p \leq i < q \wedge p \leq j < q : X.i = X_j)$$

- the segment is ascending:

$$\mathcal{A}.p.q = (\forall i, j : p \leq i \leq j < q : X.i \leq X.j)$$

- the segment contains at most 60 zeros:

$$\mathcal{A}.p.q = (\#i : p \leq i < q : X.i = 0) \leq 60$$

$\mathcal{A}$'s Properties

(0) $\mathcal{A}$ holds for the empty segment:

$$\mathcal{A}.p.p$$

(1) $\mathcal{A}$ is prefix-closed:

$$\mathcal{A}.p.q \Rightarrow (\forall s : p \leq s \leq q : \mathcal{A}.p.s)$$

(2) $\mathcal{A}$ is postfix-closed:

$$\mathcal{A}.p.q \Rightarrow (\forall i : p \leq s \leq q : \mathcal{A}.s.q)$$

Shortest Segment Problems

$$
\begin{array}{l}
|[\\
\textbf{con } N : \mathit{int}\ \{N \geq 0\};\ \ X : \textbf{array } [0..N)\ \textbf{of } \mathit{int}; \\
\textbf{var } r : \mathit{int}; \\
\mathit{minseg} \\
\{r = (\textbf{min } p, q : 0 \leq p \leq q \leq N \wedge \mathcal{A}.p.q :\ \ q - p)\} \\
]|
\end{array}
$$

Examples

- the segment must contain values 0, 1 and 2:

$$\mathcal{A}.p.q = (\exists i, j, k : p \leq i, j, k < q : X.i = 0 \wedge X.j = 1 \wedge X.k = 2)$$

- the segment contains at least 60 zeros:

$$\mathcal{A}.p.q = (\# i : p \leq i < q : X.i = 0) \geq 60$$

$\mathcal{A}$'s Properties

(0') $\neg\mathcal{A}$ holds for the empty segment:

$$\neg\mathcal{A}.p.p$$

(1') $\neg\mathcal{A}$ is prefix-closed:

$$\neg\mathcal{A}.p.q \Rightarrow (\forall s : p \leq s \leq q : \neg\mathcal{A}.p.s)$$

(2') $\neg\mathcal{A}$ is postfix-closed:

$$\neg\mathcal{A}.p.q \Rightarrow (\forall i : p \leq s \leq q : \neg\mathcal{A}.s.q)$$

Remarks

- $\mathcal{A}$ satisfies (0), (1) and (2) $\quad\equiv\quad$ $\neg\mathcal{A}$ satisfies (0'), (1') and (2').
- If we find a solution to *maxseg* for predicates that satisfy (0) and (1), then we can obtain a solution for predicates that satisfy (0) and (2).
- If we find a solution to *minseg* for predicates that satisfy $(0')$ and $(1')$, then we can obtain a solution for predicates that satisfy $(0')$ and $(2')$.

Solving Longest Segment Problems

We derive a program scheme for solving

$$
\begin{array}{l}
|[\\
\textbf{con } N : int \; \{N \geq 0\}; \; X : \textbf{array } [0..N) \textbf{ of } int; \\
\textbf{var } r : int; \\
maxseg \\
\{r = (\textbf{max } p, q : 0 \leq p \leq q \leq N \wedge \mathcal{A}.p.q : \; q - p)\} \\
]|
\end{array}
$$

where $\mathcal{A}$ satisfies (0) and (1):

(0) $\mathcal{A}$ holds for the empty segment: $\mathcal{A}.p.p$

(1) $\mathcal{A}$ is prefix-closed: $\mathcal{A}.p.q \Rightarrow (\forall s : p \leq s \leq q : \mathcal{A}.p.s)$

Slope Search

For the post condition:

$$R : r = (\mathbf{max}\ p, q : 0 \leq p \leq q \leq N \wedge \mathcal{A}.p.q :\ q - p)$$

since $q - p$ is ascending in q and descending in p, we may define

$$G.a.b = (\mathbf{max}\ p, q : a \leq p \leq N \wedge b \leq q \leq N \wedge \mathcal{A}.p.q :\ q - p)$$

Thus,

$$R : r = G.0.0$$

What are the invariants?

Invariants

$$P_0: \quad r\ \mathbf{max}\ G.a.b = G.0.0$$
$$P_1: \quad 0 \le a \le b \le N$$

which can be initialized by

$$a, b, r = 0, 0, 0$$

When does the loop terminate?

Deriving the guard

$$
\begin{array}{ll}
 & G.a.N \\
= & \quad \{ \text{ definition of } G \text{ } \} \\
 & (\mathbf{max}\ p : a \le p \le N \wedge \mathcal{A}.p.N : N - p) \\
= & \quad \{ \text{ assume } \mathcal{A}.a.N,\ N - p \text{ is descending in } p \text{ } \} \\
 & N - a
\end{array}
$$

Hence,

$$P_0 \wedge \underline{b = N \wedge \mathcal{A}.a.b} \;\Rightarrow\; R(r := r\ \mathbf{max}\ (N - a))$$

How to determine a condition under which b may be increased?

Calculating a recursive definition for G

$$
\begin{array}{rcll}
G.a.b & = & G.a.(b+1)\ \mathbf{max}\ (b-a), & \text{if } \mathcal{A}.a.b \\
 & = & G.(a+1).b, & \text{if } \neg\mathcal{A}.a.b
\end{array}
$$

Can you derive this definition?

What is the final program?

The final program

$$
\begin{array}{l}
|[\\
\textbf{var } a, b : \mathit{int}; \\
a, b, r := 0, 0, 0; \\
\textbf{do } b \neq N \vee \neg \mathcal{A}.a.b \rightarrow \\
\quad \textbf{if } \mathcal{A}.a.b \rightarrow r := r \ \textbf{max} - (b - a);\ \ b := b + 1 \\
\quad [] \neg \mathcal{A}.a.b \rightarrow a := a + 1 \\
\quad \textbf{fi} \\
\textbf{od}; \\
r := r \ \textbf{max} \ (N - a) \\
]|
\end{array}
$$

Solving Shortest Segment Problems

We derive a program scheme for solving

$$
\begin{array}{l}
|[\\
\textbf{con } N : int \ \{N \geq 0\}; \ X : \textbf{array } [0..N) \textbf{ of } int; \\
\textbf{var } r : int; \\
minseg \\
\{r = (\textbf{min } p, q : 0 \leq p \leq q \leq N \wedge \mathcal{A}.p.q : \ q - p)\} \\
]|
\end{array}
$$

where $\mathcal{A}$ satisfies (0') and (2'):

(0') $\neg\mathcal{A}$ holds for the empty segment: $\neg\mathcal{A}.p.p$

(2') $\neg\mathcal{A}$ is postfix-closed: $\neg\mathcal{A}.p.q \Rightarrow (\forall s : p \leq s \leq q : \neg\mathcal{A}.s.q)$

Slope Search

For the post condition:

$$R : r = (\mathbf{min}\ p, q : 0 \leq p \leq q \leq N \wedge \mathcal{A}.p.q :\ \ q - p)$$

since $q - p$ is ascending in q and descending in p, we may define

$$G.a.b = (\mathbf{min}\ p, q : a \leq p \leq N \wedge b \leq q \leq N \wedge \mathcal{A}.p.q :\ \ q - p)$$

Thus,

$$R : r = G.0.0$$

What are the invariants?

Invariants

$$
\begin{aligned}
P_0 : &\quad r \;\mathbf{min}\; G.a.b = G.0.0 \\
P_1 : &\quad 0 \le a \le b \le N
\end{aligned}
$$

which can be initialized by

$$a, b, r = 0, 0, \infty$$

When does the loop terminate?

Deriving the guard

$$
\begin{array}{ll}
 & G.a.N \\
= & \quad \{ \text{ definition of } G \text{ } \} \\
 & (\mathbf{min}\ p : a \leq p \leq N \wedge \mathcal{A}.p.N : N - p) \\
= & \quad \{ \text{ assume } \neg\mathcal{A}.a.N,\ \neg\mathcal{A} \text{ is postfix-closed } \} \\
 & \infty
\end{array}
$$

Hence,

$$P_0 \wedge P_1 \wedge \underline{b = N \wedge \neg\mathcal{A}.a.b} \ \Rightarrow\ R$$

How to determine a condition under which b may be increased?

Calculating a recursive dtefinition for G

$$\begin{array}{lcll} G.a.b & = & G.a.(b+1) & \text{if} \neg \mathcal{A}.a.b \\ & = & G.(a+1).b \ \mathbf{min} \ (b-a), & \text{if } \mathcal{A}.a.b \end{array}$$

Can you derive this definition?

What is the final program?

The final program

$$
\begin{array}{l}
|[\\
\textbf{var } a, b : \mathit{int}; \\
a, b, r := 0, 0, \infty; \\
\textbf{do } b \neq N \vee \mathcal{A}.a.b \rightarrow \\
\quad \textbf{if } \neg\mathcal{A}.a.b \rightarrow b := b + 1 \\
\quad [\,]\ \mathcal{A}.a.b \rightarrow r := r \textbf{ min } (b - a);\ a := a + 1 \\
\quad \textbf{fi} \\
\textbf{od}; \\
]|
\end{array}
$$

Shortest Segment with at Least Two Zeros

$|[$

con $N : int \; \{N \geq 0\}$; X : **array** $[0..N)$ **of** int;

var $r : int$;

at-least-two-zeros

$\{r = (\textbf{min}\; p, q : 0 \leq p \leq q \leq N \wedge \mathcal{A}.p.q : \; q - p)\}$

$]|$

where

$$\mathcal{A}.a.b = (\#i : a \leq i < b : X.i = 0) \geq 2$$

Refining $\mathcal{A}$

Since $\neg\mathcal{A}$ hods for the empty segment and $\neg\mathcal{A}$ is postfix-closed, we can use our obtained program scheme. We refine $\mathcal{A}$ by introducing the invariant:

$$Q : c = (\#i : a \leq i < b : X.i = 0)$$

Then

$$\mathcal{A}.a.b \equiv c \geq 2$$
$$\neg\mathcal{A}.a.b \equiv c < 2$$

We obtain the following program:

$$
\begin{array}{l}
|[\\
\textbf{var}\ a, b, \underline{c} : \mathit{int}; \\
a, b, r := 0, 0, \infty;\ \ \underline{c := 0;} \\
\textbf{do}\ b \neq N \vee \underline{c \geq 2} \rightarrow \\
\quad \textbf{if}\ \underline{c < 2} \rightarrow \underline{\textbf{if}\ X.b = 0\ \rightarrow c := c + 1\ []\ X.b \neq 0 \rightarrow \mathit{skip}};\\
\qquad\qquad\qquad b := b + 1 \\
\quad []\ \underline{c \geq 2} \rightarrow \underline{\textbf{if}\ X.a = 0\ \rightarrow c := c - 1\ []\ X.a \neq 0 \rightarrow \mathit{skip}};\\
\qquad\qquad\qquad r := r\ \textbf{min}\ (b - a);\ \ a := a + 1 \\
\quad \textbf{fi} \\
\textbf{od}; \\
]|
\end{array}
$$

Report 2

Solve Problem 5 to Problem 8, and send your report by email to

hu@mist.i.u-tokyo.ac.jp

no later than

July 31st, 2004.

The subject of your email should be MSP #2.

THANK YOU!