Programming Languages: Imperative Program Construction
Practicals 6: Loop Constuction II

Shin-Cheng Mu

Autumn Term, 2024

1. Recall the maximum segment sum problem. What if we want to compute the maximum sum of *non-empty* segments?
   (a) How would you write the specification? Does the specification still make sense with N being constrained only by $0 \leqslant N$?
   (b) Derive a program solving the problem, assuming $-\infty$ to be the identity of $(\uparrow)$.

2. Let $n : Nat$. Trying explaining why the following "splitting off" step is wrong by trying out the range calculation.

$$\begin{array}{ll} & \langle \Sigma i : n+1 \leqslant i < n+1 : f[i] \rangle \\ = & \quad \{ \text{ splitting off } i = n \} \\ & \langle \Sigma i : n+1 \leqslant i < n : f[i] \rangle + f[n] \ . \end{array}$$

3. Derive a solution for:

$$\begin{array}{l} \textbf{con } N : Int\{N \geqslant 0\}; a : \textbf{array } [0..N) \textbf{ of } Int \\ \textbf{var } r : Int \\ S \\ \{ r = \langle \uparrow i, j : 0 \leqslant i < j < N : a[i] - a[j] \rangle \} \ . \end{array}$$

4. Derive a solution for:

$$\begin{array}{l} \textbf{con } N : Int\{N \geqslant 1\}; a : \textbf{array } [0..N) \textbf{ of } Int \\ \textbf{var } r : Int \\ S \\ \{ r = \langle \#i, j : 0 \leqslant i < j < N : a[i] \times a[j] \geqslant 0 \rangle \} \ . \end{array}$$

5. Consider again the maximum segment sum problem and its derivation in the handouts. Since the computation of r requires the value of the subterm $\langle \uparrow p : 0 \leqslant p \leqslant n+1 : sum\ p\ (n+1) \rangle$, some would think it makes more sense to use the following loop invariant $P_0 \wedge P_1 \wedge Q$, where

$$\begin{array}{l} P_0 \equiv r = \langle \uparrow p\ q : 0 \leqslant p \leqslant q \leqslant n : sum\ p\ q \rangle) \ , \\ P_1 \equiv s = \langle \uparrow p : 0 \leqslant p \leqslant n+1 : sum\ p\ (n+1) \rangle \ , \\ Q \equiv 0 \leqslant n \leqslant N \ . \end{array}$$

   What situation could you run into, if you try to construct a program using the invariant above? What if the array is non-empty, that is, $1 \leqslant N$?