Programming Languages: Imperative Program Construction
Practicals 8: Case Studies

Shin-Cheng Mu

Autumn Term, 2024

1. For $r, b : Int$, verify the following program.

   $\{0 \leqslant r < b \wedge even\ b\}$
   $b := b \,/\, 2$
   if $r < b \rightarrow skip$
   $\mid\ b \leqslant r \rightarrow r := r - b$
   fi
   $\{0 \leqslant r < b\}$

2. Derive a $O(\log N)$ algorithm for computing square root:

   con $N : Int\ \{0 \leqslant N\}$
   var $x : Int$
   $?$
   $\{x^2 \leqslant N < (x+1)^2\}$,

   by introducing a variable y and use $P_0 \wedge P_1$ as the invariant, where

   $P_0 \equiv x^2 \leqslant N < (x+y)^2$,
   $P_1 \equiv 0 \leqslant k \wedge y = 2^k$.

3. Derive, again, a $O(\log N)$ algorithm for computing square root:

   con $N : Int\ \{0 \leqslant N\}$
   var $x : Int$
   $?$
   $\{x^2 \leqslant N < (x+1)^2\}$.

   (a) This time, construct an algorithm using binary search. What Φ will you use? Does your program rely on the fact that x^2 is monotonic on x (that is, $x \geqslant y \Rightarrow x^2 \geqslant y^2$)?

   (b) Knowing that $x \geqslant y \Rightarrow x^2 \geqslant y^2$, after the loop terminates, what can you conclude in addition to $x^2 \leqslant N < (x+1)^2$?