CIS 575. Introduction to Algorithm Analysis
Material for February 12, 2024

Constructing a Provably Correct Program

1 Constructing a Provably Correct Program

In the previous note we saw how to prove the correctness of an already given program. We shall now show how to **construct** a program **together** with its **correctness proof**.

Recall that in the very first course note we developed a specification for finding the (floor) integer square root:

Precondition $x \geq 0$

Postcondition $y^2 \leq x \ \wedge \ (y+1)^2 > x$

Let us now **construct** a program that meets this specification. The key idea is to let the loop invariant Φ be one of the conjuncts from the postcondition:

$$\Phi : y^2 \leq x$$

since then **correctness** can be achieved by letting the loop guard G be the *negation* of the other conjunct:

$$G : (y+1)^2 \leq x$$

To **establish** the invariant, we may use $y \leftarrow 0$ (which works since $x \geq 0$); to **maintain** it, we may use $y \leftarrow y+1$ (since then $(y')^2 = (y+1)^2 \leq x$). We have arrived at the program

$y \leftarrow 0$
while $(y+1)^2 \leq x$
  $y \leftarrow y+1$

which obviously **terminates** and which **by construction** is a correct implementation of the specification of (floor) integer square root. A more sophisticated development (as you may have seen in CIS301) will allow for a much more efficient implementation.