

A Heuristic for Finding a Loop Invariant

1. Start with conjuncts that express the bounds on the loop's "control" variables.
2. Then add conjuncts that express the relationships among the other variables involved in the loop. Often the loop invariant is similar in structure to the loop's postcondition. Remember:
 - the invariant must hold whenever control reaches the guards on any iteration
 - the invariant and the negation of the disjunction of the guards ($\neg BB$) must imply the postcondition.
3. Check that the loop invariant holds the first time control reaches the loop. If not, go back to step 2.
4. Check that the loop invariant is preserved by the body of the loop. If not, go back to step 2.

The heuristics we study later for the derivation of loops will give more insight into the construction of loop invariants for existing programs.

A Heuristic for Finding a Bound Function

1. Rewrite the termination condition (i.e., $\neg BB$) as an integer expression that is constantly equal to 0 (or some positive constant). Sometimes the variable bounds or other conjuncts of the loop's invariant may be needed to do this.
2. Weaken the relation generated above to form a new relation that is greater than or equal to 0 at initialization and after each iteration. Often you may need to examine the commands that change the "control" variables of the loop. This new relation is called the variant condition.
3. Obtain the bound function directly from the variant condition.