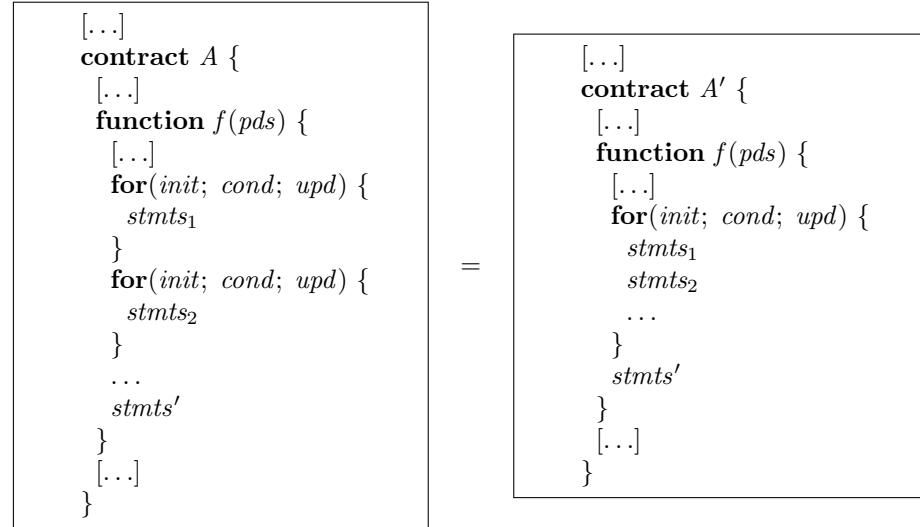

Rule 0.22 *(Combine Multiple Loops into One)*


where

Multiple loops iterate over the same range with identical *init*, *cond*, and *upd*;
stmts₁, stmts₂, ..., stmts_n represent the bodies of the separate loops;
init, *cond*, and *upd* are the loop initialization, condition, and update expressions;
pds are the parameter declarations of function *f*;
stmts' represents statements following the loops.

provided

All loops iterate over the same range or collection;
The loop bodies *stmts₁, stmts₂, ...* do not have dependencies that require sequential execution across iterations;
No loop body modifies variables that affect the iteration behavior of other loops;
Combining the loops does not introduce race conditions or ordering issues;
The statements in each loop body are independent of the execution order relative to other loop bodies within the same iteration.

Invariant:

Let *s_i* and *s'_i* be the initial state of *A* and *A'*, respectively.
Let *s_f* and *s'_f* be the state reached by *A* and *A'*, respectively, after *A.f()* and *A'.f()* are executed from *s_i* and *s'_i*, respectively.
Then, the coupling invariant is

$$\forall s_i, s'_i . (s_i = s'_i) \rightarrow (s_f = s'_f)$$
