

499 Let  $a$  and  $b$  be binary interactive variables. Define

$loop = \mathbf{if } b \mathbf{ then } loop \mathbf{ else } ok \mathbf{ fi}$

Add a time variable according to any reasonable measure, and then express

$b := \perp \parallel loop$

as an equivalent program but without using  $\parallel$ .

After trying the question, scroll down to the solution.

§ The left process owns  $b$ . Variable  $a$  could belong to either process; let's give it to the left process also. Let assignment take time 1.

$$\begin{aligned}
& b := \perp \parallel \text{loop} && \text{definition of assignment and loop with time} \\
= & b t' = \perp \wedge a t' = a t \wedge t' = t+1 \parallel \mathbf{if } b t \mathbf{ then } t := t+1. \text{ loop } \mathbf{else } \mathbf{ok } \mathbf{fi} \\
& \text{on the left, use context; on the right, use definition of loop again and ok} \\
= & b(t+1) = \perp \wedge a(t+1) = a t \wedge t' = t+1 \\
& \parallel \mathbf{if } b t \mathbf{ then } t := t+1. \mathbf{if } b t \mathbf{ then } t := t+1. \text{ loop } \mathbf{else } t' = t \mathbf{ fi } \mathbf{else } t' = t \mathbf{ fi} \\
& \text{substitution law} \\
= & b(t+1) = \perp \wedge a(t+1) = a t \wedge t' = t+1 \\
& \parallel \mathbf{if } b t \mathbf{ then if } b(t+1) \mathbf{ then } t := t+2. \text{ loop } \mathbf{else } t' = t+1 \mathbf{ fi } \mathbf{else } t' = t \mathbf{ fi} \\
& \text{definition of } \parallel \\
= & \exists tL, tR. \quad b(t+1) = \perp \wedge a(t+1) = a t \wedge tL = t+1 \\
& \wedge \mathbf{if } b t \mathbf{ then if } b(t+1) \mathbf{ then } tR = t+2. \text{ loop } \mathbf{else } tR = t+1 \mathbf{ fi } \mathbf{else } tR = t \mathbf{ fi} \\
& \wedge t' = tL \uparrow tR \\
& \text{context: } b(t+1) = \perp \\
= & \exists tL, tR. \quad b(t+1) = \perp \wedge a(t+1) = a t \wedge tL = t+1 \\
& \wedge \mathbf{if } b t \mathbf{ then if } \perp \mathbf{ then } tR = t+2. \text{ loop } \mathbf{else } tR = t+1 \mathbf{ fi } \mathbf{else } tR = t \mathbf{ fi} \\
& \wedge t' = tL \uparrow tR \\
& \text{case base} \\
= & \exists tL, tR. \quad b(t+1) = \perp \wedge a(t+1) = a t \wedge tL = t+1 \\
& \wedge \mathbf{if } b t \mathbf{ then } tR = t+1 \mathbf{ else } tR = t \mathbf{ fi} \\
& \wedge t' = tL \uparrow tR \\
& \text{factor out } tR = \\
= & \exists tL, tR. \quad b(t+1) = \perp \wedge a(t+1) = a t \wedge tL = t+1 \\
& \wedge tR = \mathbf{if } b t \mathbf{ then } t+1 \mathbf{ else } t \mathbf{ fi} \\
& \wedge t' = tL \uparrow tR \\
& \text{one-point for both } tL \text{ and } tR \\
= & b(t+1) = \perp \wedge a(t+1) = a t \wedge t' = (t+1) \uparrow \mathbf{if } b t \mathbf{ then } t+1 \mathbf{ else } t \mathbf{ fi} && \text{factor in } (t+1) \uparrow \\
= & b(t+1) = \perp \wedge a(t+1) = a t \wedge t' = \mathbf{if } b t \mathbf{ then } (t+1) \uparrow (t+1) \mathbf{ else } (t+1) \uparrow t \mathbf{ fi} \\
& \text{simplify the two } \uparrow \\
= & b(t+1) = \perp \wedge a(t+1) = a t \wedge t' = \mathbf{if } b t \mathbf{ then } t+1 \mathbf{ else } t+1 \mathbf{ fi} && \text{case idempotent} \\
= & b(t+1) = \perp \wedge a(t+1) = a t \wedge t' = t+1 && \text{context} \\
= & b t' = \perp \wedge a t' = a t \wedge t' = t+1 \\
= & b := \perp
\end{aligned}$$