415 The notation **do** P **while** b **od** has been used as a loop construct that is executed as follows. First P is executed; then b is evaluated, and if b is $\top$, execution is repeated, and if b is $\bot$, execution is finished. Define **do** P **while** b **od** by construction and induction axioms.

After trying the question, scroll down to the solution.

§ Using recursive time,

$$t' \ge t \ \Longleftarrow\ \textbf{do } P \textbf{ while } b \textbf{ od}$$

$$P.\ \ \textbf{if } b \textbf{ then } t:=t+1.\ \ \textbf{do } P \textbf{ while } b \textbf{ od else } ok \textbf{ fi} \ \Longleftarrow\ \textbf{do } P \textbf{ while } b \textbf{ od}$$

$$\forall\sigma, \sigma'\cdot\ t' \ge t \land (P.\ \ \textbf{if } b \textbf{ then } t:=t+1.\ \ D \textbf{ else } ok \textbf{ fi}) \ \Leftarrow\ D$$
$$\Longrightarrow\ \forall\sigma, \sigma'\cdot\ \textbf{do } P \textbf{ while } b \textbf{ od} \ \Leftarrow\ D$$

Recursive time does not count iterations of the **do** *P* **while** *b* **od** loop because it makes the first iteration free. To count iterations we need a different placement of the time increase, namely

$$t' \ge t \ \Longleftarrow\ \textbf{do } P \textbf{ while } b \textbf{ od}$$

$$t:=t+1.\ \ P.\ \ \textbf{if } b \textbf{ then do } P \textbf{ while } b \textbf{ od else } ok \textbf{ fi} \ \Longleftarrow\ \textbf{do } P \textbf{ while } b \textbf{ od}$$

$$\forall\sigma, \sigma'\cdot\ t' \ge t \land (t:=t+1.\ \ P.\ \ \textbf{if } b \textbf{ then } D \textbf{ else } ok \textbf{ fi}) \ \Leftarrow\ D$$
$$\Longrightarrow\ \forall\sigma, \sigma'\cdot\ \textbf{do } P \textbf{ while } b \textbf{ od} \ \Leftarrow\ D$$