Let all variables be integer. Add recursive time. Any way you can, find a fixed-point of

(a) \[ \text{walk} \equiv \begin{cases} \text{if } i \geq 0 \text{ then } i := i - 2. \text{ walk. } i := i + 1 \text{ else ok fi} \end{cases} \]

§ Putting \( t := t + 1 \) before just the first call of \( \text{walk} \) is enough, though we could put it before both calls. If the \( \equiv \) had been \( \Leftarrow \), we could prove \( i' = i \land t' \leq t + 2^i \) as follows.

Proof by cases:

i \geq 0 \land (i := i - 2, i' = i \land t' \leq t + 2^i, i := i + 1, i' = i \land t' \leq t + 2^i, i := i + 1) \quad \text{substitution}

\Rightarrow i' = i \land t' \leq t + 2^i

But the question did not say \( \Leftarrow \), so we haven't found the fixed-point asked for.

(b) \[ \text{crawl} \equiv \begin{cases} \text{if } i \geq 0 \text{ then } i := i - 1. \text{ crawl. } i := i + 2. \text{ crawl. } i := i - 1 \text{ else ok fi} \end{cases} \]

§ Putting \( t := t + 1 \) before just the first call of \( \text{crawl} \) is enough, though we could put it before both calls. Here are two answers.

if \( i \geq 0 \) then \( t' = \infty \) else ok fi

if \( i \geq 0 \) then \( t := \infty \) else ok fi

I'll check the second one.

if \( i \geq 0 \) then \( i := i - 1. \) \( t := t + 1. \) \( \text{if } i \geq 0 \text{ then } t := \infty \) else ok fi.

\( i := i + 2. \) \( \text{if } i \geq 0 \) then \( t := \infty \) else ok fi. \( i := i - 1 \) distribute \( i := i - 1 \) back into previous if else ok fi

\Rightarrow i' = i \land t' \leq t + 2^i

\exists i''', t''', i''' = i - 2 \land t''' \leq t + 2^i \land i'''' = i'' + 1 \land t'''' \leq t + 2^i + 1 \land i''' = i''' + 1

\Rightarrow i' = i \land t' \leq t + 2^i

\end{document}
Now a tricky move. In the inner then-part, use context $i \geq 1$.
And in the inner else-part, use context $i < 1$.

\[ \text{else \; ok \; fi} \]

\[ = \begin{cases} \text{if } i \geq 0 \text{ then } \exists t'' \cdot \text{ if } i \geq 1 \text{ then } t'' = \infty \text{ else } t + 1 \text{ fi} \\ \text{else } t'' = \text{ if } i \geq 1 \text{ then } \infty \text{ else } t + 1 \text{ fi} \end{cases} \]
\[ \land t' = \infty \land i' = i \]
\[ \text{else \; ok \; fi} \]
\[ = \begin{cases} \text{if } i \geq 0 \text{ then } t' = \infty \land i' = i \\ \text{else \; ok \; fi} \end{cases} \]
\[ = \text{if } i \geq 0 \text{ then } t := \infty \text{ else \; ok \; fi} \]
So it's a fixed-point.

(c) \[ \text{run} = \begin{cases} \text{if \; even \; i \; then } i := i/2 \text{ else } i := i + 1 \text{ fi} \\ \text{if } i = 1 \text{ then } \text{ok else } \text{run fi} \end{cases} \]

\$\$ Without adding time, $i' = 1$ and $i \geq 1 \Rightarrow i' = 1$ are fixed-points. With time, it's difficult to say a fixed-point since it requires saying the exact execution time. If we had an implication instead of an equation, we could get away with a time bound. Recursive construction just leads to a mess, and isn't helpful. To state the exact execution time, define

\[ f = \langle i : \text{int} \; \text{if } i = 2 \text{ then } 0 \text{ else if } \text{even } i \text{ then } 1 + f(i/2) \text{ else } 1 + f(i+1) \text{ fi fi} \] 

Now we can find the following two fixed-points:

\[ i' = 1 \land t' = t + f i \]
\[ (i \geq 1 \Rightarrow i' = 1) \land t' = t + f i \]

although $f$ seems like an unfair trick.