# The Rules

Describing precisely what a programming language construct means mathematically allows you to reason about it, implement it, and understand its properties. This activity will help you explore transition semantics, a.k.a. small step semantics.

Here are the rules presented in the video.

$\langle \texttt{skip}\ , \sigma \rangle \to \langle E, \sigma \rangle$
$\langle u := t, \sigma \rangle \to \langle E, \sigma[u := \sigma(t)] \rangle$

$$\frac{\langle S_1, \sigma \rangle \to \langle S_2, \tau \rangle}{\langle S_1; S, \sigma \rangle \to \langle S_2; S, \tau \rangle}$$

$E; S \equiv S$

$\langle \texttt{if}\ \ B\ \texttt{then}\ \ S_1\ \texttt{else}\ \ S_2\ \texttt{fi}\ , \sigma \rangle \to \langle S_1, \sigma \rangle \quad \text{where } \sigma \models B$
$\langle \texttt{if}\ \ B\ \texttt{then}\ \ S_1\ \texttt{else}\ \ S_2\ \texttt{fi}\ , \sigma \rangle \to \langle S_2, \sigma \rangle \quad \text{where } \sigma \models \neg B$

$\langle \texttt{while}\ \ B\ \texttt{do}\ \ S_1\ \texttt{od}\ , \sigma \rangle \to \langle S_1; \texttt{while}\ \ B\ \texttt{do}\ \ S_1\ \texttt{od}\ , \sigma \rangle \quad \text{where } \sigma \models B$
$\langle \texttt{while}\ \ B\ \texttt{do}\ \ S_1\ \texttt{od}\ , \sigma \rangle \to \langle E, \sigma \rangle \quad \text{where } \sigma \models \neg B$