Example program proof — while rule
In this example, we prove a program assertion establishing partial correctness of a program fragment computing the factorial of an integer.

\[
\begin{align*}
\{ \text{N} \geq 0 \} \\
M := 0; \ F := 1; \\
\text{while} \ M < N \text{ do} \\
\text{begin} \ M := M+1; \ F := M \ast F \text{ end} \\
\{ F = N! \}
\end{align*}
\]

To construct the proof of this program, we need to determine an intermediate formula \(P\) that will serve as a loop invariant. This will describe the relationship of the variable values at the intermediate point noted below.

\[
\begin{align*}
\{ \text{N} \geq 0 \} \\
M := 0; \ F := 1; \\
\text{while} \ M < N \text{ do} \\
\text{begin} \ M := M+1; \ F := M \ast F \text{ end} \\
\{ F = N! \}
\end{align*}
\]

We take \(P\) to be the formula \(\{ F = M! \wedge 0 \leq M \leq N \}\). Then the proof is as follows:

1. By two applications of the axiom of assignment and the sequential execution rule

\[
\begin{align*}
\{ 1 = 0! \wedge 0 \leq N \} \ M := 0; \ F := 1 \ P \\
\text{and the pre-condition is logically equivalent to the pre-condition of the program.}
\end{align*}
\]

2. By two applications of the axiom of assignment and the sequential execution rule

\[
\begin{align*}
\{ (M+1)! = (M+1)! \wedge 0 \leq M+1 \leq N \} \ M := M+1; \ F := M \ast F \ P \\
\end{align*}
\]

3. Since \(P \wedge M < N \Rightarrow (M+1)! = (M+1)! \wedge 0 \leq M+1 \leq N\), by the rule for strengthening the pre-condition, \(P\) is an invariant for the while loop.
4. By step 3 and the while rule

\[PP\]
\[\textbf{while } M < N \textbf{ do} \]
\[\textbf{begin } M := M + 1; \ F := M \times F \textbf{ end} \]
\[ PP \land M \geq N \]

5. Since \( PP \land M \geq N \Rightarrow F = N! \), by the rule for weakening the post-condition, the program proof is complete.