X5.6 In ProTem, there is a conditional program with the following form.

if *b0* ⟦ *P0* ⟧
b1 ⟦ *P1* ⟧
...
bn ⟦ *Pn* ⟧

where *b0* , *b1* , ..., *bn* are binary expressions, and *P0* , *P1* , ..., *Pn* are programs, and there can be any positive finite number of pairs of binary expression and program. The binary expressions are evaluated in order until the first one that has value ⊤ , then the corresponding program is executed. If none of the binary expressions is ⊤ , then none of the programs is executed. Formally define this conditional program, but generalize from programs to specifications.

After trying the question, scroll down to the solution.

§ A partial formalization is:

if *b0* **then** *P0*
else if *b1* **then** *P1*
...
else if *bn* **then** *Pn* **fi fi** ... **fi**

To finish formalizing, the binary expressions and specifications are parameterized as $b\ i$ and $P\ i$. Quantifying over $0,..n+1$,

$$(\forall i\cdot\ b\ i\ \wedge\ (\forall j\cdot\ j<i \Rightarrow \neg\ b\ j) \Rightarrow P\ i)\ \wedge\ ((\forall i\cdot\ \neg\ b\ i)\ \Rightarrow ok)$$

Maybe there's a simpler formalization.