Exercise I3 *A program π is called* totally correct *with respect to precondition A and postcondition B if the following holds: when π starts in a state where A is true then π terminates in a state where B is true.*
A program π is called partially correct *with respect to precondition A and postcondition B if the following holds: when π starts in a state where A is true and terminates then B is true after termination.*

In both cases we may assume that A and B are arithmetical formulas.
In both cases correctness assertions may be written compactly as $\{A\}\pi\{B\}$.
What follows from the incompleteness theorems about the provability of total and partial correctness assertions? What if A, B, and/or π are trivial? Provide concrete examples and explain!

If no truth constants are present in the arithmetic language, let **true** be $0 = 0$ and **false** be $0 \neq 0$. Moreover, let `skip` be the empty program, that trivially terminates without changing the state.

Consider (partial or total) correctness assertions of the form $\{\mathbf{true}\}\mathtt{skip}\{A\}$, where A is an arbitrary arithmetic sentence. By definition, any such correctness assertion is true if and only if A is true in formal arithmetic. By the incompleteness theorem there is no correct proof system for arithmetic where all true formulas are provable. Therefore we also cannot have a correct proof system where all true (partial or total) correctness assertions are provable.

An analogous argument can be made for (partial as well as total) correctness assertions of the form $\{A\}\mathtt{skip}\{\mathbf{false}\}$, since these are true if and only if $\neg A$ is true in formal arithmetic. (Note that "π starts in a state where A is true" is false if A is not true in any state.)

Note that it does not matter, whether we talk about partial or total correctness in the above examples. Moreover no appeal to undecidability is needed – only incompleteness of formal arithmetic is used.

If both, the precondition and the postcondition, are trivial then *total* correctness assertions are either trivially true or false as well. However we obtain a connection between the undecidability of the Halting Problem and incompleteness for *partial* correctness assertions in this case as follows. The partial correctness assertion $\{\mathbf{true}\}\pi\{\mathbf{false}\}$ is true if and only if the program π never terminates. Therefore we also know that not all true partial correctness assertions can be provable, since otherwise the Halting Problem were decidable.