Consider the syntax for the following simple imperative language IMP:

expressions \( e ::= v | x | e_1 + e_2 | e_1 - e_2 | e_1 * e_2 | e_1 = e_2 | e_1 <= e_2 | ! e_1 | e_1 & e_2 | e_1 || e_2 \)

values \( v \in \text{Val} ::= n \mid b \)

variables \( x \in \text{Var} \)

integers \( n \)

booleans \( b ::= \text{true} \mid \text{false} \)

commands \( c ::= \text{skip} \mid x := e \mid c_1 ; c_2 \mid \text{if} \ e \ \text{then} \ c_1 \ \text{else} \ c_2 \mid \text{while} \ e \ \text{do} \ c_1 \)

1. **Semantics.** Let \( \sigma : \text{Var} \rightarrow \text{Val} \) be a concrete store mapping variables to values.

   (a) Let \( \langle \sigma, e \rangle \downarrow v \) be the judgment form for a big-step operational semantics defining the evaluation of expressions \( e \). Define this judgment form. You may assume usual mathematical operators for integers and booleans.
(b) Let $\langle \sigma, c \rangle \downarrow \sigma'$ be the judgment form for a big-step operational semantics defining the evaluation of commands $c$. Define this judgment form.

(c) Let $\langle \sigma, c \rangle \rightarrow \langle \sigma', c' \rangle$ be the judgment form for a small-step operational semantics defining the evaluation of commands $c$. Define this judgment form. You may use the big-step semantics for evaluating expressions.
2. **Program Verification.** Prove by induction the following statement about the big-step operational semantics of IMP:

For any $e$ and any initial state $\sigma$ such that $\sigma(x)$ is even, if

$$D :: (\sigma, \textbf{while } e \textbf{ do } x := x + 2) \downarrow \sigma'$$

then $\sigma'(x)$ is even.

Make sure you state what you induct on and where you apply the induction hypothesis.
3. **Meta-Theory.** Consider the reflexive-transitive closure of the small-step relation defined as follows:

\[
\begin{align*}
\text{MSREFL} & \quad \frac{\langle \sigma, c \rangle \rightarrow^* \langle \sigma, c \rangle}{\langle \sigma, c \rangle}
\end{align*}
\]

\[
\begin{align*}
\text{MTRANS} & \quad \frac{\langle \sigma, c \rangle \rightarrow \langle \sigma', c' \rangle \quad \langle \sigma', c' \rangle \rightarrow^* \langle \sigma'', c'' \rangle}{\langle \sigma, c \rangle \rightarrow^* \langle \sigma'', c'' \rangle}
\end{align*}
\]

Prove the following stating an equivalence between big-step evaluation and small-step reduction (or rather, one side of the equivalence):

If \( M :: \langle \sigma, c \rangle \rightarrow^* \langle \sigma', \text{skip} \rangle \), then \( E :: \langle \sigma, c \rangle \Downarrow \sigma' \).

You will need to come up with a lemma that relates a single step reduction with evaluation.