Syntax and Semantics
Exercise Session 11

We next present two simplified proofs from Hans Hüttel's book *Transition and Trees*, ISBN 978-0-521-14709-5. The proofs in the book are given in the context of a more general language than `Bims`.
Notation. We shall write $s \vdash e \to v$ instead of $s \vdash e \to_E v$ to avoid confusion with the type environment E that appears in the statements of both theorems. (Recall that the E in $\to_E$ emphasizes the fact that we are dealing with a typed expression and does not refer to a type environment).

Theorem 13.4, p. 193. *(Safety of Expression) Suppose that state s agrees with the type environment E and that $E \vdash e : T$, where e is an expression and T is a type. Then $s \vdash e \to v$ and $v \in \boldsymbol{set}(T)$, where $\boldsymbol{set}(T)$ denotes the set of values corresponding to type T.*

Proof. This is a result that states a property for all terms that are well-typed. We therefore prove this theorem by induction on the depth of the derivation tree for $E \vdash e : T$.

$n = 0$: Here, there are two cases to consider.

- If $E \vdash e : T$ was concluded using $[\texttt{VAR}_{\texttt{EXP}}]$, then $e = x$ for some variable x, and since E agrees with s, we have that $s(x) = v$ for some $v \in \mathbf{set}(T)$. The desired property is now seen to hold immediately.
- Otherwise, $E \vdash e : \texttt{int}$ such that e is some numeral n, allowing us to infer $s \vdash n \to v$ where $v = \mathcal{N}[\![n]\!] \in \mathbb{Z}$. Since $\mathbb{Z} = \mathbf{set}(\texttt{int})$ and $E \vdash n : \texttt{int}$, this yields the claim.

Assume for all $j \leq n$, prove for $n + 1$:
Suppose the derivation tree for $E \vdash e : T$ has height $n + 1$, then one of the rules not covered by base cases above must have been used. We must then

consider each rule in turn. All of the cases are similar, so we give proof only for one of them here.
Suppose [$\texttt{ADD}_{EXP}$] was the last rule used. The derivation tree for $E \vdash e_1 : \texttt{int}$ has height $j_1 \leq n$ and the derivation tree for $E \vdash e_2 : \texttt{int}$ has height $j_2 \leq n$. By virtue of our induction hypothesis, we have $s \vdash e_1 \to v_1$ with $v_1 \in \mathbb{Z}$ and $s \vdash e_2 \to v_2$ with $v_2 \in \mathbb{Z}$. From the transition rule [$\texttt{PLUS}_{BS}$], we have that $s \vdash e_1 + e_2 \to v_1 + v_2$ and clearly $v_1 + v_2 \in \mathbb{Z}$ and $\mathbb{Z} = \mathbf{set}(\texttt{int})$, as we needed to prove. □

Theorem 13.9, p. 196. *(Safety of Statements) Suppose that s agrees with the type environment E and that $E \vdash S : ok$, where s is the state, S is a statement, and ok is the label that tells us a statement is well-typed. Then, if $\langle S, s\rangle \to s'$, E agrees with s'.*

Proof. In this theorem we state a property for all transitions, so here the proof proceeds by induction on the height of the derivation tree of the transition $\langle S, s\rangle \to s'$.

$n = 0$: Here, there are two cases to consider.

- If [$\texttt{SKIP}_{BS}$] was used, the conclusion is immediate, since $\langle S, s\rangle \to s$. By definition `skip` gets the label *ok*, and there is therefore no type violation.

- If [$\texttt{ASSIGN}_{BS}$] was used, we have that S is $x := e$ and that $E \vdash e : T$ and $E(x) = T$. But by Theorem 13.4 we have that $s \vdash e \to v$ with $v \in \mathbf{set}(T)$, so no type violation occurs. Moreover, it then also follows that $s' = s[x \mapsto v]$ agrees with E.

Assume for all $j \leq n$, prove for $n + 1$:
We describe only one case below where the previous theorems about the type system are invoked. The remaining cases are all similar or simpler.

- Consider [$\texttt{while} - \texttt{true}_{BS}$]. The premises tell us that $\langle S, s\rangle \to s''$ for some s'' and that $\langle \texttt{while}\ e\ \texttt{do}\ S, s''\rangle \to s'$. As $E \vdash \texttt{while}\ e\ \texttt{do}\ S : ok$, we have that $E \vdash S : ok$ and that $E \vdash e : bool$. Since $E \vdash S : ok$ and the height of the derivation tree of $\langle S, s\rangle \to s''$ is at most n, the induction hypothesis ensures that E agrees with s''. This and the fact that the height of the derivation tree of $\langle \texttt{while}\ e\ \texttt{do}\ S, s''\rangle \to s'$ is at most n, in turn, allow us to apply the induction hypothesis to infer that E is in agreement with s', thus yielding the claim. □

This is the solution for Exercise 3:
Please note that we split the derivation tree into 3 trees because of space issues. The first rule used is $[\texttt{COMP}_{\texttt{STM}}]$ as our statement consists of two statements. Each statement must be well-typed and get the label ok in order for the composed statement to be ok. Please recall that $E(i) = \mathsf{int}$.

$$[\texttt{COMP}_{\texttt{STM}}]\frac{[\texttt{ASS}_{\texttt{STM}}]\dfrac{\vdots}{E \vdash i := \underline{10} : \mathsf{ok}} \quad [\texttt{WHILE}_{\texttt{STM}}]\dfrac{\vdots}{E \vdash \texttt{while}\ i > \underline{0}\ \texttt{do}\ i = i - \underline{1}\ :\ \mathsf{ok}}}{E \vdash i := \underline{10};\ \texttt{while}\ i > \underline{0}\ \texttt{do}\ i := i - \underline{1} : \mathsf{ok}}$$

The second tree proves that the first statement is well-typed.

$$[\texttt{ASS}_{\texttt{STM}}]\frac{[\texttt{VAR}_{\texttt{EXP}}]\dfrac{E(i) = \texttt{int}}{E \vdash i : \texttt{int}} \qquad [\texttt{NUM}_{\texttt{EXP}}]E \vdash \underline{10} : \texttt{int}}{E \vdash i := \underline{10} : \mathsf{ok}}$$

The third tree proves that the while statement is well-typed. Since both statements are well-typed, the composition of the statements are well-typed. Therefore, we have no type violations.

$$[\texttt{WHILE}_{\texttt{STM}}]\frac{[\texttt{GRT}_{\texttt{EXP}}]\dfrac{[\texttt{VAR}_{\texttt{EXP}}]\dfrac{E(i) = \texttt{int}}{E \vdash i : \texttt{int}} \quad [\texttt{NUM}_{\texttt{EXP}}]E \vdash \underline{0} : \texttt{int}}{E \vdash i > \underline{0}\ :\ \mathsf{bool}} \quad [\texttt{ASS}_{\texttt{STM}}]\frac{[\texttt{VAR}_{\texttt{EXP}}]\dfrac{E(i) = \texttt{int}}{E \vdash i : \texttt{int}} \qquad [\texttt{SUB}_{\texttt{EXP}}]\dfrac{[\texttt{VAR}_{\texttt{EXP}}]\dfrac{E(i) = \texttt{int}}{E \vdash i : \texttt{int}} \quad [\texttt{NUM}_{\texttt{EXP}}]E \vdash \underline{1} : \mathsf{int}}{E \vdash i - \underline{1} : \texttt{int}}}{E \vdash i := i - \underline{1}\ :\ \mathsf{ok}}}{E \vdash \texttt{while}\ i > \underline{0}\ \texttt{do}\ i := i - \underline{1}\ :\ \mathsf{ok}}$$