CIS 705 — Programming Languages — Spring 2010

Assignment 5

Model Answers

Exercise 1

Define a predicate $P(v)$ on values by: $P(v)$ iff, if v is closed, then v is a normal form. We use structural induction on values to prove that, for all values v, $P(v)$.

(Unit Constant) We must show that $P(\mathsf{unit})$. It will suffice to show that unit is a normal form, and this follows by the inversion lemma for the evaluation relation.

(Natural Number Constant) Suppose $n \in \mathbb{N}$. We must show that $P(n)$. It will suffice to show that n is a normal form, and this follows by the inversion lemma for the evaluation relation.

(Successor Constant) We must show $P(\mathsf{succ})$. It will suffice to show that succ is a normal form, and this follows by the inversion lemma for the evaluation relation.

(Predecessor Constant) We must show $P(\mathsf{pred})$. It will suffice to show that pred is a normal form, and this follows by the inversion lemma for the evaluation relation.

(Conditional Constant for Type T) Suppose T is a type. We must show $P(\mathsf{if}[T])$. It will suffice to show that $\mathsf{if}[T]$ is a normal form, and this follows by the inversion lemma for the evaluation relation.

(Conditional Constant for Type T with One Argument) Suppose T is a type, v is a value, and assume the inductive hypothesis, $P(v)$. We must show $P(\mathsf{if}[T]\,v)$. Suppose $\mathsf{if}[T]\,v$ is closed. We must show that $\mathsf{if}[T]\,v$ is a normal form. Since $\mathsf{if}[T]\,v$ is closed, it follows that v is closed. Suppose, toward a contradiction, that $\mathsf{if}[T]\,v$ is not a normal form, so that $\mathsf{if}[T]\,v \to t$, for some closed term t. By the inversion lemma for the evaluation relation, all but two of the evaluation rules obviously don't apply to $\mathsf{if}[T]\,v \to t$, and so there are two cases to consider.

(E-App1) Suppose $t = t'\,v$, for some closed term t' such that $\mathsf{if}[T] \to t'$. But the inversion lemma for the evaluation relation shows that this is impossible—contradiction.

(E-App2) Suppose $t = \mathsf{if}[T]\,t'$, for some closed term t' such that $v \to t'$. But, since $P(v)$ and v is closed, it follows that v is a normal form—contradiction.

Since we obtained a contradiction in both cases, we have an overall contradiction. Thus $\mathsf{if}[T]\,v$ is a normal form.

(Conditional Constant for Type T with Two Arguments) Suppose T is a type, v_1 and v_2 are values, and assume the inductive hypothesis, $P(v_1)$ and $P(v_2)$. We must show $P(\mathsf{if}[T]\,v_1\,v_2)$. Suppose $\mathsf{if}[T]\,v_1\,v_2$ is closed. We must show that $\mathsf{if}[T]\,v_1\,v_2$ is a normal form. Since $\mathsf{if}[T]\,v_1\,v_2$ is closed, it follows that v_1 and v_2 are closed. Suppose, toward a contradiction, that $\mathsf{if}[T]\,v_1\,v_2$ is not a normal form, so that $\mathsf{if}[T]\,v_1\,v_2 \to t$, for some closed term t. By the inversion lemma for the evaluation relation, all but two of the evaluation rules obviously don't apply to $\mathsf{if}[T]\,v_1\,v_2 \to t$, and so there are two cases to consider.

(E-App1) Suppose $t = t'\,v_2$, for some closed term t' such that $\mathsf{if}[T]\,v_1 \rightarrow t'$. By the inversion lemma for the evaluation relation, all but two of the evaluation rules obviously don't apply to $\mathsf{if}[T]\,v_1 \rightarrow t'$, and so there are two subcases to consider.

(E-App1) Suppose $t' = t''\,v_1$, for some closed term t'' such that $\mathsf{if}[T] \rightarrow t''$. But the inversion lemma for the evaluation relation shows that this is impossible—contradiction.

(E-App2) Suppose $t' = \mathsf{if}[T]\,t''$, for some closed term t'' such that $v_1 \rightarrow t''$. But, since $P(v_1)$ and v_1 is closed, it follows that v_1 is a normal form—contradiction.

Since we obtained a contradiction in both cases, we have an overall contradiction.

(E-App2) Suppose $t = \mathsf{if}[T]\,v_1\,t'$, for some closed term t' such that $v_2 \rightarrow t'$. But, since $P(v_2)$ and v_2 is closed, it follows that v_2 is a normal form—contradiction.

Since we obtained a contradiction in both cases, we have an overall contradiction. Thus $\mathsf{if}[T]\,v_1\,v_2$ is a normal form.

(Abstraction) Suppose x is a variable, T is a type and t is a term. We must show that $P(\lambda x : T.\,t)$. Suppose $\lambda x : T.\,t$ is closed. We must show that $\lambda x : T.\,t$ is a normal form, and this follows by the inversion lemma for the evaluation relation.

Exercise 2

Define a predicate $P(t, t')$ between closed terms by: $P(t, t')$ iff, for all closed terms t'', if $t \rightarrow t''$, then $t' = t''$. It will suffice to show that, for all closed terms t and t', if $t \rightarrow t'$, then $P(t, t')$. We proceed by induction on the evaluation relation.

(E-App1) Suppose t_1, t_1' and t_2 are closed terms, and $t_1 \rightarrow t_1'$, and assume the inductive hypothesis, $P(t_1, t_1')$. We must show that $P(t_1\,t_2,\ t_1'\,t_2)$. Suppose u is a closed term, and $t_1\,t_2 \rightarrow u$. We must show that $t_1'\,t_2 = u$. Since t_1 is not a normal form, it is not a value, by Exercise 1. Because $\mathsf{if}[T]\,0\,v_1$, for a closed value v_1, $\mathsf{if}[T]\,(n+1)\,v_1$, for $n \in \mathbb{N}$ and a closed value v_1, succ, pred and $\lambda x : T.t$, for a variable x, type T and term t such that $\mathsf{FV}(t) \subseteq \{x\}$, are all closed values, the inversion lemma for the evaluation relation tells us that only (E-App1) applies to $t_1\,t_2 \rightarrow u$, so that $u = t_1''\,t_2$, for some closed term t_1'' such that $t_1 \rightarrow t_1''$. By the inductive hypothesis, $P(t_1, t_1')$, it follows that $t_1' = t_1''$. Thus $t_1'\,t_2 = t_1''\,t_2 = u$.

(E-App2) Suppose v_1 is a closed value, t_2 and t_2' are closed terms, and $t_2 \rightarrow t_2'$, and assume the inductive hypothesis, $P(t_2, t_2')$. We must show that $P(v_1\,t_2,\ v_1\,t_2')$. Suppose u is a closed term, and $v_1\,t_2 \rightarrow u$. We must show that $v_1\,t_2' = u$. Since v_1 is a closed value, it is a normal form, by Exercise 1. And, since t_2 is not a normal form, it is not a value, by Exercise 1. Furthermore n, for $n \in \mathbb{N}$, 0 and $n + 1$, for $n \in \mathbb{N}$, are values, and thus, by the inversion lemma for the evaluation relation, only (E-App2) applies to $v_1\,t_2 \rightarrow u$, so that $u = v_1\,t_2''$ for a closed term t_2'' such that $t_2 \rightarrow t_2''$. By the inductive hypothesis, $P(t_2, t_2')$, it follows that $t_2' = t_2''$. Thus $v_1\,t_2' = v_1\,t_2'' = u$.

(E-AppIfZero) Suppose T is a type, and v_1 and v_2 are closed values. We must show that $P(\mathsf{if}[T]\,0\,v_1\,v_2,\ v_1\,\mathsf{unit})$. Suppose u is a closed term, and $\mathsf{if}[T]\,0\,v_1\,v_2 \rightarrow u$. We must show that

$v_1\,\mathsf{unit} = u$. Since 0 and v_1 are closed values, we have that $\mathsf{if}[T]\,0\,v_1$ is a closed value. And, because $\mathsf{if}[T]\,0\,v_1$ and v_2 are closed values, they are normal forms, by Exercise 1. Thus, by the inversion lemma for the evaluation relation, only (E-AppIfZero) applies to $\mathsf{if}[T]\,0\,v_1\,v_2 \to u$, so that $v_1\,\mathsf{unit} = u$.

(E-AppIfNonZero) Suppose T is a type, $n \in \mathbb{N}$, and v_1 and v_2 are closed values. We must show that $P(\mathsf{if}[T]\,(n+1)\,v_1\,v_2,\ v_2\,\mathsf{unit})$. Suppose u is a closed term, and $\mathsf{if}[T]\,(n+1)\,v_1\,v_2 \to u$. We must show that $v_2\,\mathsf{unit} = u$. Since $n+1$ and v_1 are closed values, we have that $\mathsf{if}[T]\,(n+1)\,v_1$ is a closed value. And, because $\mathsf{if}[T]\,(n+1)\,v_1$ and v_2 are closed values, they are normal forms, by Exercise 1. Thus, by the inversion lemma for the evaluation relation, only (E-AppIfNonZero) applies to $\mathsf{if}[T]\,(n+1)\,v_1\,v_2 \to u$, so that $v_2\,\mathsf{unit} = u$.

(E-AppSucc) Suppose $n \in \mathbb{N}$. We must show that $P(\mathsf{succ}\,n,\ n+1)$. Suppose u is a closed term, and $\mathsf{succ}\,n \to u$. We must show that $n+1 = u$. Since succ and n are closed values, they are normal forms, by Exercise 1. Thus, by the inversion lemma for the evaluation relation, only (E-AppSucc) applies to $\mathsf{succ}\,n \to u$, so that $n+1 = u$.

(E-AppPredZero) We must show that $P(\mathsf{pred}\,0,\ 0)$. Suppose u is a closed term, and $\mathsf{pred}\,0 \to u$. We must show that $0 = u$. Since pred and 0 are closed values, they are normal forms, by Exercise 1. Thus, by the inversion lemma for the evaluation relation, only (E-AppPredZero) applies to $\mathsf{pred}\,0 \to u$, so that $0 = u$.

(E-AppPredNonZero) Suppose $n \in \mathbb{N}$. We must show that $P(\mathsf{pred}(n+1),\ n)$. Suppose u is closed term, and $\mathsf{pred}(n+1) \to u$. We must show that $n = u$. Since pred and $n+1$ are closed values, they are normal forms, by Exercise 1. Thus, by the inversion lemma for the evaluation relation, only (E-AppPredNonZero) applies to $\mathsf{pred}(n+1) \to u$, so that $n = u$.

(E-AppAbs) Suppose x is a variable, T is a type, t is a term such that $\mathsf{FV}(t) \subseteq \{x\}$, and v is a closed value. We must show that $P((\lambda x : T.\, t)v,\ [x \mapsto v]t)$. Suppose u is a closed term, and $(\lambda x : T.\, t)v \to u$. We must show that $[x \mapsto v]t = u$. Because $\lambda x : T.\, t$ and v are closed values, they are normal forms, by Exercise 1. Thus, by the inversion lemma for the evaluation relation, only (E-AppAbs) applies to $(\lambda x : T.\, t)v \to u$, and so $[x \mapsto v]t = u$.

Exercise 3

Define a predicate $P(\Gamma, t, T)$ on contexts, terms and types, by: $P(\Gamma, t, T)$ iff, for all types T', if $\Gamma \vdash t : T'$, then $T = T'$. It will suffice to show that, for all contexts Γ, terms t and types T, if $\Gamma \vdash t : T$, then $P(\Gamma, t, T)$, and we proceed by induction on the typing relation.

(T-Unit) Suppose Γ is a context. We must show that $P(\Gamma, \mathsf{unit}, \mathsf{Unit})$. Suppose T' is a type, and $\Gamma \vdash \mathsf{unit} : T'$. By the inversion lemma for the typing relation, we have that $T' = \mathsf{Unit}$. Thus $\mathsf{Unit} = T'$, as required.

(T-Nat) Suppose Γ is a context, and $n \in \mathbb{N}$. We must show that $P(\Gamma, n, \mathsf{Nat})$. Suppose T' is a type, and $\Gamma \vdash n : T'$. By the inversion lemma for the typing relation, we have that $T' = \mathsf{Nat}$. Thus $\mathsf{Nat} = T'$, as required.

(T-Succ) Suppose Γ is a context. We must show that $P(\Gamma, \mathsf{succ}, \mathsf{Nat} \to \mathsf{Nat})$. Suppose T' is a type, and $\Gamma \vdash \mathsf{succ} : T'$. By the inversion lemma for the typing relation, we have that $T' = \mathsf{Nat} \to \mathsf{Nat}$. Thus $\mathsf{Nat} \to \mathsf{Nat} = T'$, as required.

(T-Pred) Suppose Γ is a context. We must show that $P(\Gamma, \mathsf{pred}, \mathsf{Nat} \to \mathsf{Nat})$. Suppose T' is a type, and $\Gamma \vdash \mathsf{pred} : T'$. By the inversion lemma for the typing relation, we have that $T' = \mathsf{Nat} \to \mathsf{Nat}$. Thus $\mathsf{Nat} \to \mathsf{Nat} = T'$, as required.

(T-If) Suppose Γ is a context, and T is a type. We must show that $P(\Gamma, \mathsf{if}[T], \mathsf{Nat} \to (\mathsf{Unit} \to T) \to (\mathsf{Unit} \to T) \to T)$. Suppose T' is a type, and $\Gamma \vdash \mathsf{if}[T] : T'$. By the inversion lemma for the typing relation, we have that $T' = \mathsf{Nat} \to (\mathsf{Unit} \to T) \to (\mathsf{Unit} \to T) \to T$. Thus $\mathsf{Nat} \to (\mathsf{Unit} \to T) \to (\mathsf{Unit} \to T) \to T = T'$, as required.

(T-Var) Suppose Γ is a context, x is a variable, T is a type, and $(x, T) \in \Gamma$. We must show that $P(\Gamma, x, T)$. Suppose T' is a type, and $\Gamma \vdash x : T'$. We must show that $T = T'$. By the inversion lemma for the typing relation, it follows that $(x, T') \in \Gamma$. But $(x, T) \in \Gamma$ and Γ is a function, and thus $T = T'$.

(T-Abs) Suppose Γ is a context, x is a variable, t is a term, T_1 and T_2 are types, and $\Gamma[x \mapsto T_1] \vdash t : T_2$, and assume the inductive hypothesis, $P(\Gamma[x \mapsto T_1], t, T_2)$. We must show that $P(\Gamma, \lambda x : T_1.\, t, T_1 \to T_2)$. Suppose T' is a type, and $\Gamma \vdash \lambda x : T_1.\, t : T'$. We must show that $T_1 \to T_2 = T'$. By the inversion lemma for the typing relation, we have that $T' = T_1 \to T''$ for some type T'' such that $\Gamma[x \mapsto T_1] \vdash t : T''$. By the inductive hypothesis, $P(\Gamma[x \mapsto T_1], t, T_2)$, it follows that $T_2 = T''$. Thus $T_1 \to T_2 = T_1 \to T'' = T'$.

(T-App) Suppose Γ is a context, t_1 and t_2 are terms, T_1 and T_2 are types, $\Gamma \vdash t_1 : T_1 \to T_2$ and $\Gamma \vdash t_2 : T_1$, and assume the inductive hypothesis, $P(\Gamma, t_1, T_1 \to T_2)$ and $P(\Gamma, t_2, T_1)$. We must show that $P(\Gamma, t_1\, t_2, T_2)$. Suppose T' is a type, and $\Gamma \vdash t_1\, t_2 : T'$. We must show that $T_2 = T'$. By the inversion lemma for the typing relation, we have that there is a type T'' such that $\Gamma \vdash t_1 : T'' \to T'$ and $\Gamma \vdash t_2 : T''$. By $P(\Gamma, t_1, T_1 \to T_2)$, we have that $T_1 \to T_2 = T'' \to T'$, so that $T_2 = T'$.

Exercise 4

Lemma 4.1

(1) For all types T and U, and terms t_1, if $\emptyset \vdash \mathsf{if}[T]\, t_1 : U$, then $U = (\mathsf{Unit} \to T) \to (\mathsf{Unit} \to T) \to T$ and $\emptyset \vdash t_1 : \mathsf{Nat}$.

(2) For all types T and U, and terms t_1 and t_2, if $\emptyset \vdash \mathsf{if}[T]\, t_1\, t_2 : U$, then $U = (\mathsf{Unit} \to T) \to T$, $\emptyset \vdash t_1 : \mathsf{Nat}$ and $\emptyset \vdash t_2 : \mathsf{Unit} \to T$.

(3) For all types T and U, and terms t_1, t_2 and t_3, if $\emptyset \vdash \mathsf{if}[T]\, t_1\, t_2\, t_3 : U$, then $U = T$, $\emptyset \vdash t_1 : \mathsf{Nat}$, $\emptyset \vdash t_2 : \mathsf{Unit} \to T$ and $\emptyset \vdash t_3 : \mathsf{Unit} \to T$.

Proof.

(1) Suppose T and U are types, t_1 is a term, and $\emptyset \vdash \mathsf{if}[T]\, t_1 : U$. By the inversion lemma for the typing relation, we have that $\emptyset \vdash \mathsf{if}[T] : U' \to U$ and $\emptyset \vdash t_1 : U'$, for some type U'. Applying

the inversion lemma for the typing relation to $\emptyset \vdash \mathsf{if}[T] : U' \to U$, we have that $U' \to U = \mathsf{Nat} \to (\mathsf{Unit} \to T) \to (\mathsf{Unit} \to T) \to T$, so that $U' = \mathsf{Nat}$ and $U = (\mathsf{Unit} \to T) \to (\mathsf{Unit} \to T) \to T$. Thus $U = (\mathsf{Unit} \to T) \to (\mathsf{Unit} \to T) \to T$ and $\emptyset \vdash t_1 : U' = \mathsf{Nat}$.

(2) Suppose T and U are types, t_1 and t_2 are terms, and $\emptyset \vdash \mathsf{if}[T]\, t_1\, t_2 : U$. By the inversion lemma for the typing relation, we have that $\emptyset \vdash \mathsf{if}[T]\, t_1 : U' \to U$ and $\emptyset \vdash t_2 : U'$, for some type U'. Thus, by Part (1), we have that $U' \to U = (\mathsf{Unit} \to T) \to (\mathsf{Unit} \to T) \to T$ and $\emptyset \vdash t_1 : \mathsf{Nat}$, so that $U' = \mathsf{Unit} \to T$ and $U = (\mathsf{Unit} \to T) \to T$. Hence $U = (\mathsf{Unit} \to T) \to T$, $\emptyset \vdash t_1 : \mathsf{Nat}$ and $\emptyset \vdash t_2 : U' = \mathsf{Unit} \to T$.

(3) Suppose T and U are types, t_1, t_2 and t_3 are terms, and $\emptyset \vdash \mathsf{if}[T]\, t_1\, t_2\, t_3 : U$. By the inversion lemma for the typing relation, we have that $\emptyset \vdash \mathsf{if}[T]\, t_1\, t_2 : U' \to U$ and $\emptyset \vdash t_3 : U'$, for some type U'. Thus, by Part (2), we have that that $U' \to U = (\mathsf{Unit} \to T) \to T$, $\emptyset \vdash t_1 : \mathsf{Nat}$ and $\emptyset \vdash t_2 : \mathsf{Unit} \to T$, so that $U' = \mathsf{Unit} \to T$ and $U = T$. Hence $U = T$, $\emptyset \vdash t_1 : \mathsf{Nat}$, $\emptyset \vdash t_2 : \mathsf{Unit} \to T$ and $\emptyset \vdash t_3 : U' = \mathsf{Unit} \to T$.

□

Lemma 4.2
For all values v, if $\emptyset \vdash v : \mathsf{Nat}$, then $v = n$, for some $n \in \mathbb{N}$.

Proof. Suppose v is a value, and $\emptyset \vdash v : \mathsf{Nat}$. Because v is a value, there are eight cases to consider.

(1) Suppose $v = \mathsf{unit}$. Thus $\emptyset \vdash \mathsf{unit} : \mathsf{Nat}$. But applying the inversion lemma for the typing relation to this, we get a contradiction. Thus $v = n$, for some $n \in \mathbb{N}$.

(2) Suppose $v = n$, for some $n \in \mathbb{N}$. Then we are done.

(3) Suppose $v = \mathsf{succ}$. Thus $\emptyset \vdash \mathsf{succ} : \mathsf{Nat}$. Applying the inversion lemma for the evaluation relation to this, we get a contradiction. Thus $v = n$, for some $n \in \mathbb{N}$.

(4) Suppose $v = \mathsf{pred}$. Thus $\emptyset \vdash \mathsf{pred} : \mathsf{Nat}$. Applying the inversion lemma for the evaluation relation to this, we get a contradiction. Thus $v = n$, for some $n \in \mathbb{N}$.

(5) Suppose $v = \mathsf{if}[T]$, for some type T. Thus $\emptyset \vdash \mathsf{if}[T] : \mathsf{Nat}$. Applying the inversion lemma for the evaluation relation to this, we get a contradiction. Thus $v = n$, for some $n \in \mathbb{N}$.

(6) Suppose $v = \mathsf{if}[T]\, v_1$, for some type T and value v_1. Thus $\emptyset \vdash \mathsf{if}[T]\, v_1 : \mathsf{Nat}$. By Lemma 4.1(1), we have that Nat is a function type—contradiction. Thus $v = n$, for some $n \in \mathbb{N}$.

(7) Suppose $v = \mathsf{if}[T]\, v_1\, v_2$, for some type T and values v_1 and v_2. Thus $\emptyset \vdash \mathsf{if}[T]\, v_1\, v_2 : \mathsf{Nat}$. By Lemma 4.1(2), we have that Nat is a function type—contradiction. Thus $v = n$, for some $n \in \mathbb{N}$.

(8) Suppose $v = \lambda x : T.\, t$, for some variable x, type T and term t. Thus $\emptyset \vdash \lambda x : T.\, t : \mathsf{Nat}$. Applying the inversion lemma for the evaluation relation to this, we get a contradiction. Thus $v = n$, for some $n \in \mathbb{N}$.

□

Define a predicate $P(\Gamma, t, T)$ on contexts, terms and types, by: $P(\Gamma, t, T)$ iff, if $\Gamma = \emptyset$, then t is a closed term that is not stuck. It will suffice to show that, for all contexts Γ, terms t and types T, if $\Gamma \vdash t : T$, then $P(\Gamma, t, T)$. (To see this, suppose t is a closed term, and t is well typed. Then $\emptyset \vdash t : T$, for some type T, so that $P(\emptyset, t, T)$. Thus t is not stuck.) We proceed by induction on the typing relation.

(T-Unit) Suppose Γ is a context. We must show that $P(\Gamma, \mathsf{unit}, \mathsf{Unit})$. Suppose $\Gamma = \emptyset$. We must show that unit is a closed term that is not stuck, which follows since it's a closed value.

(T-Nat) Suppose Γ is a context, and $n \in \mathbb{N}$. We must show that $P(\Gamma, n, \mathsf{Nat})$. Suppose $\Gamma = \emptyset$. We must show that n is a closed term that is not stuck, which follows since it's a closed value.

(T-Succ) Suppose Γ is a context. We must show that $P(\Gamma, \mathsf{succ}, \mathsf{Nat} \to \mathsf{Nat})$. Suppose $\Gamma = \emptyset$. We must show that succ is a closed term that is not stuck, which follows since it's a closed value.

(T-Pred) Suppose Γ is a context. We must show that $P(\Gamma, \mathsf{pred}, \mathsf{Nat} \to \mathsf{Nat})$. Suppose $\Gamma = \emptyset$. We must show that pred is a closed term that is not stuck, which follows since it's a closed value.

(T-If) Suppose Γ is a context, and T is a type. We must show that $P(\Gamma, \mathsf{if}[T], \mathsf{Nat} \to (\mathsf{Unit} \to T) \to (\mathsf{Unit} \to T) \to T)$. Suppose $\Gamma = \emptyset$. We must show that $\mathsf{if}[T]$ is a closed term that is not stuck, which follows since it's a closed value.

(T-Var) Suppose Γ is a context, x is a variable, T is a type, and $(x, T) \in \Gamma$. We must show that $P(\Gamma, x, T)$. Suppose $\Gamma = \emptyset$. We must show that x is a closed term that is not stuck. Because $(x, T) \in \Gamma = \emptyset$, we have a contradiction. Thus x is a closed term that is not stuck.

(T-Abs) Suppose Γ is a context, x is a variable, t is a term, T_1 and T_2 are types, and $\Gamma[x \mapsto T_1] \vdash t : T_2$, and assume the inductive hypothesis, $P(\Gamma[x \mapsto T_1], t, T_2)$. We must show that $P(\Gamma, \lambda x : T_1 .\, t, T_1 \to T_2)$. Suppose $\Gamma = \emptyset$. We must show that $\lambda x : T_1 .\, t$ is a closed term that is not stuck. Since $\emptyset[x \mapsto T_1] = \Gamma[x \mapsto T_1] \vdash t : T_2$, the Free Variables Lemma tells us that $\mathsf{FV}(t) \subseteq \mathsf{dom}(\emptyset[x \mapsto T_1]) = \{x\}$. Thus $\lambda x : T_1 .\, t$ is a closed value, so that it is a closed term that's not stuck.

(T-App) Suppose Γ is a context, t_1 and t_2 are terms, T_1 and T_2 are types, $\Gamma \vdash t_1 : T_1 \to T_2$ and $\Gamma \vdash t_2 : T_1$, and assume the inductive hypothesis, $P(\Gamma, t_1, T_1 \to T_2)$ and $P(\Gamma, t_2, T_1)$. We must show that $P(\Gamma, t_1\, t_2, T_2)$. Suppose $\Gamma = \emptyset$. We must show that $t_1\, t_2$ is a closed term that's not stuck. Since $P(\Gamma, t_1, T_1 \to T_2)$, we have that t_1 is a closed term that's not stuck, and since $P(\Gamma, t_2, T_1)$, we have that t_2 is a closed term that's not stuck. Because t_1 and t_2 are closed, it follows that $t_1\, t_2$ is closed. Thus it remains to show that $t_1\, t_2$ is not stuck.

If t_1 is not a normal form, then there is a closed term t_1' such that $t_1 \to t_1'$, so that $t_1\, t_2 \to t_1'\, t_2$, showing that $t_1\, t_2$ is not a normal form, and thus is not stuck. Otherwise, we have that t_1 is a value. And $\emptyset \vdash t_1 : T_1 \to T_2$.

If t_2 is not a normal form, then there is a closed term t_2' such that $t_2 \to t_2'$, so that (since t_1 is a value) $t_1\, t_2 \to t_1\, t_2'$, showing that $t_1\, t_2$ is not a normal form, and thus is not stuck. Otherwise, we have that t_2 is a value. And $\emptyset \vdash t_2 : T_1$.

Because t_1 is a value, there are eight cases to consider.

(1) Suppose $t_1 = \mathsf{unit}$. Thus $\emptyset \vdash \mathsf{unit} : T_1 \to T_2$. But applying the inversion lemma for the typing relation to this, we get a contradiction. Thus $t_1\, t_2$ is not stuck.

(2) Suppose $t_1 = n$, for some $n \in \mathbb{N}$. Thus $\emptyset \vdash n : T_1 \to T_2$. But applying the inversion lemma for the typing relation to this, we get a contradiction. Thus $t_1\, t_2$ is not stuck.

(3) Suppose $t_1 = \mathsf{succ}$. Thus $\emptyset \vdash \mathsf{succ} : T_1 \to T_2$. Applying the inversion lemma for the evaluation relation to this, we get that $T_1 = \mathsf{Nat}$. Since $\emptyset \vdash t_2 : T_1$, we have that $\emptyset \vdash t_2 : \mathsf{Nat}$. By Lemma 4.2, it follows that $t_2 = n$, for some $n \in \mathbb{N}$. Thus $t_1\, t_2 = \mathsf{succ}\, n \to n + 1$, showing that $t_1\, t_2$ is not a normal form, and thus is not stuck.

(4) Suppose $t_1 = \mathsf{pred}$. Thus $\emptyset \vdash \mathsf{pred} : T_1 \to T_2$. Applying the inversion lemma for the evaluation relation to this, we get that $T_1 = \mathsf{Nat}$. Since $\emptyset \vdash t_2 : T_1$, we have that $\emptyset \vdash t_2 : \mathsf{Nat}$. By Lemma 4.2, it follows that $t_2 = n$, for some $n \in \mathbb{N}$. There are two subcases to consider.

- Suppose $n = 0$. Then $t_1\, t_2 = \mathsf{pred}\, 0 \to 0$, showing that $t_1\, t_2$ is not a normal form, and thus is not stuck.
- Suppose $n = m + 1$ for some $m \in \mathbb{N}$. Then $t_1\, t_2 = \mathsf{pred}(m + 1) \to m$, showing that $t_1\, t_2$ is not a normal form, and thus is not stuck.

(5) Suppose $t_1 = \mathsf{if}[T]$, for some type T. Then $t_1\, t_2 = \mathsf{if}[T]\, t_2$ is a value (since t_2 is a value) and thus is not stuck.

(6) Suppose $t_1 = \mathsf{if}[T]\, v$, for some type T and value v. Thus $t_1\, t_2 = \mathsf{if}[T]\, v\, t_2$ is a value (since v and t_2 are values), and thus is not stuck.

(7) Suppose $t_1 = \mathsf{if}[T]\, v_1\, v_2$, for some type T and values v_1 and v_2. Then $\emptyset \vdash \mathsf{if}[T]\, v_1\, v_2 : T_1 \to T_2$. By Lemma 4.1(2), it follows that $\emptyset \vdash v_1 : \mathsf{Nat}$. Thus, by Lemma 4.2, we have that $v_1 = n$, for some $n \in \mathbb{N}$. There are two subcases to consider.

- Suppose $n = 0$. Since v_2 and t_2 are closed values, we have that $t_1\, t_2 = \mathsf{if}[T]\, v_1\, v_2\, t_2 = \mathsf{if}[T]\, 0\, v_2\, t_2 \to v_2\, \mathsf{unit}$, showing that $t_1\, t_2$ is not a normal form, and thus that $t_1\, t_2$ is not stuck.
- Suppose $n = m + 1$ for some $m \in \mathbb{N}$. Since v_2 and t_2 are closed values, we have that $t_1\, t_2 = \mathsf{if}[T]\, v_1\, v_2\, t_2 = \mathsf{if}[T]\, (m + 1)\, v_2\, t_2 \to t_2\, \mathsf{unit}$, showing that $t_1\, t_2$ is not a normal form, and thus that $t_1\, t_2$ is not stuck.

(8) Suppose $t_1 = \lambda x : T.\, t$, for some variable x, type T and term t. Since t_1 is closed, it follows that $\mathsf{FV}(t) \subseteq \{x\}$. And t_2 is a closed value, so that $t_1\, t_2 = (\lambda x : T.\, t)t_2 \to [x \mapsto t_2]t$, showing that $t_1\, t_2$ is not a normal form, and thus is not stuck.

Exercise 5

Define a predicate $P(t, t')$ between closed terms by: $P(t, t')$ iff, for all types T, if $\emptyset \vdash t : T$, then $\emptyset \vdash t' : T$. It will suffice to show that, for all closed terms t and t', if $t \to t'$, then $P(t, t')$. We proceed by induction on the evaluation relation.

(E-App1) Suppose t_1, t_1' and t_2 are closed terms, and $t_1 \rightarrow t_1'$, and assume the inductive hypothesis, $P(t_1, t_1')$. We must show that $P(t_1\, t_2,\, t_1'\, t_2)$. Suppose T is a type, and $\emptyset \vdash t_1\, t_2 : T$. We must show that $\emptyset \vdash t_1'\, t_2 : T$. By the inversion lemma for the typing relation, we have that $\emptyset \vdash t_1 : T' \rightarrow T$ and $\emptyset \vdash t_2 : T'$, for some type T'. By the inductive hypothesis, $P(t_1, t_1')$, it follows that $\emptyset \vdash t_1' : T' \rightarrow T$. Since $\emptyset \vdash t_1' : T' \rightarrow T$ and $\emptyset \vdash t_2 : T'$, we have that $\emptyset \vdash t_1'\, t_2 : T$.

(E-App2) Suppose v_1 is a closed value, t_2 and t_2' are closed terms, and $t_2 \rightarrow t_2'$, and assume the inductive hypothesis, $P(t_2, t_2')$. We must show that $P(v_1\, t_2,\, v_1\, t_2')$. Suppose T is a type, and $\emptyset \vdash v_1\, t_2 : T$. We must show that $\emptyset \vdash v_1\, t_2' : T$. By the inversion lemma for the typing relation, we have that $\emptyset \vdash v_1 : T' \rightarrow T$ and $\emptyset \vdash t_2 : T'$, for some type T'. By the inductive hypothesis, $P(t_2, t_2')$, it follows that $\emptyset \vdash t_2' : T'$. Since $\emptyset \vdash v_1 : T' \rightarrow T$ and $\emptyset \vdash t_2' : T'$, we have that $\emptyset \vdash v_1\, t_2' : T$.

(E-AppIfZero) Suppose T is a type, and v_1 and v_2 are closed values. We must show that $P(\mathsf{if}[T]\, 0\, v_1\, v_2,\, v_1\, \mathsf{unit})$. Suppose U is a type, and $\emptyset \vdash \mathsf{if}[T]\, 0\, v_1\, v_2 : U$. We must show that $\emptyset \vdash v_1\, \mathsf{unit} : U$. By Lemma 4.1(3), we have that $U = T$ and $\emptyset \vdash v_1 : \mathsf{Unit} \rightarrow T$. But $\emptyset \vdash \mathsf{unit} : \mathsf{Unit}$, and thus $\emptyset \vdash v_1\, \mathsf{unit} : T = U$.

(E-AppIfNonZero) Suppose T is a type, $n \in \mathbb{N}$, and v_1 and v_2 are closed values. We must show that $P(\mathsf{if}[T]\, (n+1)\, v_1\, v_2,\, v_2\, \mathsf{unit})$. Suppose U is a type, and $\emptyset \vdash \mathsf{if}[T]\, (n+1)\, v_1\, v_2 : U$. We must show that $\emptyset \vdash v_2\, \mathsf{unit} : U$. By Lemma 4.1(3), $U = T$ and $\emptyset \vdash v_2 : \mathsf{Unit} \rightarrow T$. But $\emptyset \vdash \mathsf{unit} : \mathsf{Unit}$, and thus $\emptyset \vdash v_2\, \mathsf{unit} : T = U$.

(E-AppSucc) Suppose $n \in \mathbb{N}$. We must show that $P(\mathsf{succ}\, n,\, n+1)$. Suppose T is a type, and $\emptyset \vdash \mathsf{succ}\, n : T$. We must show that $\emptyset \vdash n+1 : T$. By the inversion lemma for the typing relation, we have that $\emptyset \vdash \mathsf{succ} : T' \rightarrow T$ and $\emptyset \vdash n : T'$, for some type T'. Applying the inversion lemma for the typing relation to $\emptyset \vdash \mathsf{succ} : T' \rightarrow T$, we have that $T = \mathsf{Nat}$. Thus $\emptyset \vdash n+1 : \mathsf{Nat} = T$.

(E-AppPredZero) We must show that $P(\mathsf{pred}\, 0,\, 0)$. Suppose T is a type, and $\emptyset \vdash \mathsf{pred}\, 0 : T$. We must show that $\emptyset \vdash 0 : T$. By the inversion lemma for the typing relation, we have that $\emptyset \vdash \mathsf{pred} : T' \rightarrow T$ and $\emptyset \vdash 0 : T'$, for some type T'. Applying the inversion lemma for the typing relation to $\emptyset \vdash \mathsf{pred} : T' \rightarrow T$, we have that $T = \mathsf{Nat}$. Thus $\emptyset \vdash 0 : \mathsf{Nat} = T$.

(E-AppPredNonZero) Suppose $n \in \mathbb{N}$. We must show that $P(\mathsf{pred}(n+1),\, n)$. Suppose T is a type, and $\emptyset \vdash \mathsf{pred}(n+1) : T$. We must show that $\emptyset \vdash n : T$. By the inversion lemma for the typing relation, we have that $\emptyset \vdash \mathsf{pred} : T' \rightarrow T$ and $\emptyset \vdash (n+1) : T'$, for some type T'. Applying the inversion lemma for the typing relation to $\emptyset \vdash \mathsf{pred} : T' \rightarrow T$, we have that $T = \mathsf{Nat}$. Thus $\emptyset \vdash n : \mathsf{Nat} = T$.

(E-AppAbs) Suppose x is a variable, T is a type, t is a term such that $\mathsf{FV}(t) \subseteq \{x\}$, and v is a closed value. We must show that $P((\lambda x : T.\, t)v,\, [x \mapsto v]t)$. Suppose U is a type, and $\emptyset \vdash (\lambda x : T.\, t)v : U$. We must show that $\emptyset \vdash [x \mapsto v]t : U$. By the inversion lemma for the typing relation, we have that $\emptyset \vdash \lambda x : T.\, t : U' \rightarrow U$ and $\emptyset \vdash v : U'$, for some type U'. Applying the inversion lemma for the typing relation to $\emptyset \vdash \lambda x : T.\, t : U' \rightarrow U$, we have that $T = U'$ and $\emptyset[x \mapsto T] \vdash t : U$. Thus $\emptyset[x \mapsto U'] \vdash t : U$. Since $\emptyset[x \mapsto U'] \vdash t : U$ and $\emptyset \vdash v : U'$, the Typing of Substitutions Lemma tells us that $\emptyset \vdash [x \mapsto v]t : U$.