Assignment on Lambda Calculus and Types

For your reference, we give the formalization of the untyped λ-calculus and the simply-typed λ-calculus in the appendix at the end of this document.

1. In the untyped λ-calculus, does the following property hold?

   $$\text{For any terms } M \text{ and } N, \text{ if } M \to N, \text{ then } fv(M) = fv(N).$$

   If it holds, just answer yes; otherwise, please give a counterexample (that is, instantiate M and N such that $M \to N$ but $fv(M) \neq fv(N)$).

2. In the untyped λ-calculus, the *span* of a term is the minimal number of variables needed to write a term. It turns out that β-reduction can increase the span of a term. To show this, find a closed term M such that $M \to^* \lambda x.\ \lambda y.\ M\ (x\ y)$.

3. In this problem we add "let-bindings" to the untyped λ-calculus.

   Syntax (the syntax for Values is unchanged):

   $$(\mathsf{Terms}) \quad M \quad ::= \quad \ldots \mid \mathsf{let}\ x = M\ \mathsf{in}\ M$$

   New reduction rules (note v is a value):

   $$\frac{}{\mathsf{let}\ x = v\ \mathsf{in}\ M \to M[v/x]}\ (\text{LETV})$$

   $$\frac{M_1 \to M_1'}{\mathsf{let}\ x = M_1\ \mathsf{in}\ M_2 \to \mathsf{let}\ x = M_1'\ \mathsf{in}\ M_2}\ (\text{LET})$$

   (a) The (LETV) rule uses substitution, but since we have extended the syntax of terms, the definition of substitution should be extended as well. Give the definition of the substitution $M[N/x]$ when M is in the form of let-bindings.

   (b) Reduce the following term to a normal form.

   $$\begin{array}{l} \mathsf{let}\ c = \lambda n.\ \lambda m.\ \lambda f.\ \lambda x.\ n\ f\ (m\ f\ x)\ \mathsf{in} \\ \quad (\mathsf{let}\ b = \lambda f.\ \lambda x.\ f\ (f\ x)\ \mathsf{in} \\ \quad\quad (\mathsf{let}\ a = \lambda f.\ \lambda x.\ f\ x\ \mathsf{in} \\ \quad\quad\quad (c\ b\ a))) \end{array}$$

   You can choose any reduction strategy. Please do not skip steps.

4. In this problem we add the option types to the simply-typed λ-calculus. We can use $\textsf{None}$ and $\textsf{Some}$ to construct terms of the option type, just like `None` and `Some` in Coq. Intuitively, $\textsf{None}$ represents a dummy element (i.e. there is no meaningful element), $\textsf{Some}\ M$ means that there is a meaningful element M, and $\textsf{get}\ M$ gives us the meaningful element contained in M of the option type.

   Syntax:

$$
\begin{array}{lccl}
\textsf{(Types)} & \tau & ::= & \ldots \mid \textsf{option}\ \tau \\
\textsf{(Terms)} & M & ::= & \ldots \mid \textsf{None} \mid \textsf{Some}\ M \mid \textsf{get}\ M \\
\textsf{(Values)} & v & ::= & \ldots \mid \textsf{None} \mid \textsf{Some}\ v
\end{array}
$$

   New reduction rules:

$$
\frac{M \to M'}{\textsf{Some}\ M \to \textsf{Some}\ M'}\ (\textsc{some}) \qquad \frac{M \to M'}{\textsf{get}\ M \to \textsf{get}\ M'}\ (\textsc{get-m})
$$

$$
\frac{}{\textsf{get}\ (\textsf{Some}\ M) \to M}\ (\textsc{get-some}) \qquad \frac{}{\textsf{get None} \to \textsf{get None}}\ (\textsc{get-none})
$$

   (a) Give 3 appropriate new typing rules, one for each new form of term. Note that your rules should ensure the preservation and progress theorems (though you don't need to show their proofs).

   (b) Consider each of the following questions in isolation. Answer yes or no.

      i. Suppose we remove the above (SOME) rule.
      Does the preservation theorem still hold?
      Does the progress theorem still hold?

      ii. Suppose we add the following rule.

$$
\frac{}{\textsf{get}\ v \to \textsf{get}\ v}\ (\textsc{get-v})
$$

      Does the preservation theorem still hold?
      Does the progress theorem still hold?

      iii. Suppose we change the above (GET-SOME) rule to the following (GET-SOME') rule.

$$
\frac{}{\textsf{get}\ (\textsf{Some}\ v) \to v}\ (\textsc{get-some'})
$$

      Does the preservation theorem still hold?
      Does the progress theorem still hold?

      iv. Suppose we change the above (GET-NONE) rule to the following (GET-NONE') rule.

$$
\frac{}{\textsf{get None} \to \textsf{None}}\ (\textsc{get-none'})
$$

      Does the preservation theorem still hold?
      Does the progress theorem still hold?

Appendix

A The Untyped λ-Calculus

Syntax:

$$\begin{array}{rrcl} \textsf{(Terms)} & M & ::= & x \mid \lambda x.\ M \mid M\ N \\ \textsf{(Values)} & v & ::= & \lambda x.\ M \end{array}$$

Reduction rules:

$$\frac{}{(\lambda x.\ M)\ N \ \rightarrow\ M[N/x]} \qquad \frac{M \rightarrow M'}{\lambda x.\ M \ \rightarrow\ \lambda x.\ M'}$$

$$\frac{M \rightarrow M'}{M\ N \ \rightarrow\ M'\ N} \qquad \frac{N \rightarrow N'}{M\ N \ \rightarrow\ M\ N'}$$

Substitution:

$$\begin{array}{l} x[N/x] = N \\ y[N/x] = y \\ (M\,N)[N'/x] = (M[N'/x])\,(N[N'/x]) \\ (\lambda x.\,M)[N/x] = \lambda x.\,M \\ (\lambda y.\,M)[N/x] = \lambda y.\,(M[N/x]), \quad \text{where } y \notin \mathit{fv}(N) \\ (\lambda y.\,M)[N/x] = \lambda z.\,(M[z/y])[N/x], \quad \text{where } y \in \mathit{fv}(N) \text{ and } z \text{ fresh} \end{array}$$

Free variables:

$$\mathit{fv}(x) = \{x\} \qquad \mathit{fv}(M\,N) = \mathit{fv}(M) \cup \mathit{fv}(N) \qquad \mathit{fv}(\lambda x.\,M) = \mathit{fv}(M) - \{x\}$$

Zero-or-more steps:

$$\begin{array}{lll} M \rightarrow^{0} M' & \text{iff} & M = M' \\ M \rightarrow^{k+1} M' & \text{iff} & \exists M''.\ M \rightarrow M'' \wedge M'' \rightarrow^{k} M' \\ M \rightarrow^{*} M' & \text{iff} & \exists k.\ M \rightarrow^{k} M' \end{array}$$

Normal form: a term containing no redex.
Closed term: a term containing no free variables.

B The Simply-Typed λ-Calculus

Syntax (here T denotes the base type):

$$\begin{array}{rrcl} \textsf{(Types)} & \tau & ::= & \mathrm{T} \mid \tau \rightarrow \tau \\ \textsf{(Terms)} & M & ::= & x \mid \lambda x : \tau.\ M \mid M\ N \\ \textsf{(Values)} & v & ::= & \lambda x : \tau.\ M \\ \textsf{(Contexts)} & \Gamma & ::= & \bullet \mid \Gamma, x : \tau \end{array}$$

Reduction rules:

$$\frac{}{(\lambda x:\tau.\ M)\ N\ \rightarrow\ M[N/x]} \qquad \frac{M \rightarrow M'}{\lambda x:\tau.\ M\ \rightarrow\ \lambda x:\tau.\ M'}$$

$$\frac{M \rightarrow M'}{M\ N\ \rightarrow\ M'\ N} \qquad \frac{N \rightarrow N'}{M\ N\ \rightarrow\ M\ N'}$$

Typing rules:

$$\frac{}{\Gamma, x:\tau \vdash x:\tau} \qquad \frac{\Gamma, x:\tau \vdash M:\tau'}{\Gamma \vdash (\lambda x:\tau.\ M):\tau \rightarrow \tau'}$$

$$\frac{\Gamma \vdash M:\tau \rightarrow \tau' \qquad \Gamma \vdash N:\tau}{\Gamma \vdash M\ N:\tau'}$$

Preservation:

For any M, M' and τ, if $\bullet \vdash M:\tau$ and $M \rightarrow M'$, then $\bullet \vdash M':\tau$.

Progress:

For any M and τ, if $\bullet \vdash M:\tau$, then either $M \in \mathsf{Values}$ or $\exists M'.\ M \rightarrow M'$.