COMP6463 (Semester 2, 2013)
Assignment for Typed Lambda Calculus

Due: Wednesday, 9 October 2013, 5pm EST

Submission information: Submit your assignment either

- on paper, to a lecturer of the course, or to the office in the ground floor of the RSISE building, or
- as a PDF file (scanned handwritten papers are acceptable, provided they are readable) via email to jeremy.dawson@anu.edu.au

by the due date.

Please note: some, but not all, questions about unification or finding a principal type specifically ask you to show *all* the steps used in applying an algorithm discussed in lectures. For other questions, simply show clearly how you obtained your answers.

Question 1 (10 marks) Find a most general unifier for the set of pairs

$$E = \{(\alpha \to \gamma \to \delta, \alpha' \to \beta'), (\alpha \to \beta \to \gamma, \beta' \to \alpha')\}$$

. Show *all* the steps of the unification algorithm shown in lectures.

Question 2 (25 marks) For each term below, use the Principal Type Algorithm (PT) to determine whether it is typeable in λ_{Cu}. If it is typeable, give a principal type and a principal deduction. Show *all* the steps used in applying the PT Algorithm and in finding any necessary unifiers.

(a) $\lambda f \lambda z.\ f\,(f\,z)$

(b) $\lambda x \lambda y.\ x$

(c) $(\lambda f \lambda z.\ f\,(f\,z))\ (\lambda x \lambda y.\ x)$ (note, this (a) applied to (b))

Question 3 (15 marks) One way of encoding natural numbers in λ-calculus is via the so-called *Church numerals*. A number n is encoded as the λ-term $\lambda f \lambda x. f^n x$, where $f^n x$ means f applied n times to x.

For example, 0 is represented as $\lambda f \lambda x. x$; 1 is represented as $\lambda f \lambda x. f\,x$, 2 is represented as $\lambda f \lambda x. f\,(f\,x)$ and so on.

Write $C_1 = \lambda f \lambda x. f\,x$, $C_2 = \lambda f \lambda x. f\,(f\,x)$, $C_3 = \lambda f \lambda x. f\,(f\,(f\,x))$, etc. Thus $C_i\ f = f^i$ for any i.

Since $f^{mn} = (f^m)^n$, we can define multiplication of these Church numerals by (T for "times")

$$T\ C_m\ C_n\ f = C_m(C_n f), \qquad \text{that is,} \qquad T = \lambda m \lambda n \lambda f.\ m\,(n\,f)$$

Likewise, since $f^{m+n}(x) = f^m(f^n(x))$ we can define addition of Church numerals by (P for "plus")

$$P\ C_m\ C_n\ f\ x = C_m\,f\,(C_n\,f\,x) \qquad \text{that is,} \qquad P = \lambda m \lambda n \lambda f \lambda x.\ m\,f\,(n\,f\,x)$$

Since T and P are both functions which take two Church numerals as arguments and produce a third Church numeral:

(a) Find the principal types for T and for P (using the definitions of T and P above)

(b) Can the types of T and of P be unified to a common type? What is it?

(c) How does this compare with the type you might expect T and P to have, given that they are binary operators on Church numerals? Why?

Question 4 (10 marks) Consider the term $(\lambda x.\ x)\ x$

(a) In this term, identify the free and bound occurrences of x

(b) Give a typing derivation for this term (Hint: you may need to use α-equivalence)

Question 5 (20 marks) Assuming the substitution lemma,

(a) prove that the basic β-reduction step, $(\lambda x.M)\ N \longrightarrow_\beta M[x := N]$, preserves the type: that is, if $(\lambda x.M)\ N : \tau$ then $M[x := N] : \tau$

(b) from this, prove that any single step β-reduction of a term (including where the reduction takes place at a subterm of the term) preserves the type of the term

Question 6 (20 marks) We had an example in lectures of $M \longrightarrow_\beta N$, where N is typeable but M is not. In that example, the explanation was that M is an abstraction which ignored its argument, and N is an untypeable argument. Here is an example with a different explanation of how this can happen.

(a) Show that $(\lambda x.\ x\ x)\ (\lambda y.\ y)$ is not typeable.

(b) Find its β-normal form.

(c) Show that its β-normal form is typeable.

(d) Can you explain how β-reduction changes an untypeable term to a typeable one in this case?