Types and λ-calculus

Problem Sheet 4

This week, Questions 3 and 6 will be marked.

* 1. Justify each of the following conversions $M \approx N$ by finding a common reduct P, i.e. such that $M \rhd^* P$ and $N \rhd^* P$.

(a) $(\lambda x.x)y \approx (\lambda xy.x)\ y\ z$

(b) $(\lambda x.M)N \approx M[N/x]$

(c) $\mathsf{fix}\ (\underline{\mathsf{const}}\ \underline{1}) \approx M\ (\underline{\mathsf{const}}\ \mathsf{pred}\ 2)$

(d) $z\ (\underline{\mathsf{const}}\ \underline{\mathsf{id}}\ \underline{\mathsf{div}})\ \underline{\mathsf{div}} \approx z\ \underline{\mathsf{id}}\ (\underline{\mathsf{const}}\ \underline{\mathsf{div}}\ \underline{\mathsf{id}})$

* 2. Define $\underline{\mathsf{Y}} = \lambda f.(\lambda x.f\ (x\ x))(\lambda x.f\ (x\ x))$.

Show that $\underline{\mathsf{Y}}$ is also a fixed point combinator, i.e for all terms M:

$$\underline{\mathsf{Y}} M \approx M (\underline{\mathsf{Y}} M)$$

** 3. Prove Lemma 8.1 of the notes, i.e. show all of the following:

Reflexivity For all M: $M \approx M$.

Symmetry For all M, N: $M \approx N$ implies $N \approx M$.

Transitivity For all M, N and P: $M \approx P$ and $P \approx N$ implies $M \approx N$.

Compatibility For all M, N and $C[]$: if $M \approx N$ then $C[M] \approx C[N]$.

There is no need for any induction. For compatibility, you will need to use 9(b) from the previous problem sheet.

* 4. At the end of the lecture we defined addition, $\underline{\mathsf{add}}$, as:

$$\mathsf{fix}\,(\lambda f x y.\,\mathsf{ifz}\; x\; y\; (\mathsf{S}\; (f\; (\mathsf{pred}\; x)\; y)))$$

Give a complete reduction sequence from $\underline{\mathsf{add}}\ \underline{2}\ \underline{3}$ to $\underline{5}$.

* 5.

(a) Prove that $\underline{\mathsf{add}}$ satisfies the following equation:

$$\underline{\mathsf{add}} \;\approx\; \lambda x y.\,\mathsf{ifz}\; x\; y\; (\mathsf{S}\; (\underline{\mathsf{add}}\; (\mathsf{pred}\; x)\; y))$$

(b) Prove that $\underline{\mathsf{add}}$ satisfies the following equations. Induction is not necessary.

$$\underline{\mathsf{add}}\ \underline{0}\ \underline{m} \;\approx\; \underline{m} \qquad \text{and} \qquad \underline{\mathsf{add}}\ \underline{(n+1)}\ \underline{m} \;\approx\; \mathsf{S}\; (\underline{\mathsf{add}}\ \underline{n}\ \underline{m})$$

Hint: In practice, you nearly always want to replace an occurrence of $\underline{\mathsf{add}}$ with the right-hand-side of the equation in (a), rather than by its actual definition (and the same can be said for any recursive function defined using "the recipe").

** 6. Use fix to define the recursive triangular number function: using the "recipe", give a combinator $\underline{\mathsf{Tri}}$ that satisfies:

$$\underline{\mathsf{Tri}}\ \underline{0} \;\approx\; \underline{0} \qquad \text{and} \qquad \underline{\mathsf{Tri}}\ \underline{(n+1)} \;\approx\; \underline{\mathsf{add}}\ \underline{(n+1)}\ (\underline{\mathsf{Tri}}\ \underline{n})$$

Convince yourself that $\underline{\mathsf{Tri}}\ \underline{2} \;\approx\; \underline{3}$ (this is obvious if you believe that your implementation of $\underline{\mathsf{Tri}}$ really satisfies the given equations).

** 7. Define multiplication, i.e. construct a term $\underline{\mathsf{mult}}$ that satisfies the following specification:

$$\underline{\mathsf{mult}}\ \underline{0}\ \underline{m} \;\approx\; \underline{0} \qquad \text{and} \qquad \underline{\mathsf{mult}}\ \underline{n+1}\ \underline{m} \;\approx\; \underline{\mathsf{add}}\ \underline{m}\ (\underline{\mathsf{mult}}\ \underline{n}\ \underline{m})$$

Convince yourself that $\underline{\mathsf{mult}}\ \underline{2}\ \underline{2} \;\approx\; \underline{4}$.

** 8. Prove that if $M \approx N$ and N is a normal form, then $M \rhd^* N$.

Therefore, we now know that e.g. $\underline{\mathsf{Tri}}\ \underline{2} \rhd^* \underline{3}$ and $\underline{\mathsf{mult}}\ \underline{2}\ \underline{2} \rhd^* \underline{4}$, so these definitions actually *compute* an output given an input.