Assignment 5

COMP 302 Programming Languages and Paradigms
Prakash Panangaden

Solutions

The solutions to the programming questions are on the course web page.

Question 3 [20 points] Please derive a principal type of the following function using the sort of informal reasoning I have shown in class:

\[
\text{let } S = \text{fun } x \rightarrow (\text{fun } y \rightarrow (\text{fun } z \rightarrow (x \ z) \ (y \ z)))
\]

Solution We can immediately see that S is some kind of function that takes x as input and outputs a new function (\text{fun } y \rightarrow \ldots). Let us assign x to a fresh type \( \alpha \). The output function is taking as input y and outputs a new function (\text{fun } z \rightarrow \ldots). Let \( \beta \) be the type we assign to y. Finally, let us assign a fresh type \( \gamma \) to z, and \( \delta \) to the output of the function that takes z as input, that is \((xz)(yz) : \delta\). Then, for the moment, we have that S is of the following type: \( \alpha \rightarrow (\beta \rightarrow (\gamma \rightarrow \delta)) \). From \((x \ z)\) we see that x is applied to z, and we can thus infer that x is some kind of function. Therefore \( \alpha = \epsilon \rightarrow \theta \) for some new \( \epsilon \) and \( \theta \). But x takes z as input and we know \( z : \gamma \), thus it must be \( \epsilon = \gamma \) and \( x : \gamma \rightarrow \theta \). Looking at \((y \ z)\) and using a similar reasoning, we get \( y : \beta = \gamma \rightarrow \eta \) for some new \( \eta \). By looking at \((x \ z)\) \((y \ z)\) we see that the output of \((xz)\) is applied to the output of \((yz)\), that is \((xz)\) must output a function which takes \((yz)\) as input. Therefore \( \theta \) must be a function type whose input is of type \( \eta \), that is \( \theta = \eta \rightarrow \zeta \) for some new \( \zeta \). Now we have \((xz) : \gamma \rightarrow (\eta \rightarrow \zeta)\) and \((yz) : \gamma \rightarrow \eta\), thus \((xz)(yz)\), to which we initially assigned type \( \delta \), must be of type \( \zeta \). Finally, putting everything together, we deduce that the type of S must be:

\[
x : (\gamma \rightarrow (\eta \rightarrow \zeta)) \rightarrow (y : (\gamma \rightarrow \eta) \rightarrow (z : \gamma \rightarrow \zeta))
\]

Solution by Giulia Alberini

Question 4 [15 points] Suppose we have a programming language called G-flat in which the type int is a subtype of float. Explain why \( \text{int} \rightarrow \text{int} \) is not a subtype of \( \text{float} \rightarrow \text{float} \). Please do not ramble on and on. Anything more than 10 lines will be ignored, so if you plan on writing 17 pages make sure that the correct answer is in the first 10 lines.
Solution. If I want a function of type \( \text{float} \rightarrow \text{float} \) it means that I plan on providing this function with values of type float. If I am given a function of type \( \text{int} \rightarrow \text{int} \) I have a function that accepts integers. But float is not a subtype of int which means that the function will not be able to deal with the input I will provide it. Thus I cannot use a function of type \( \text{int} \rightarrow \text{int} \) in place of a function of type \( \text{float} \rightarrow \text{float} \).