15 Types (nk480)

(a) Recall that in constructive logic, logical negation is defined using implication and falsehood as \( \neg A \triangleq A \supset \bot \).

(i) Does \( A \supset \neg \neg A \)? If so, give a simply-typed lambda-term corresponding to this implication.

(ii) Does \( \neg \neg A \supset A \)? If so, give a simply-typed lambda-term corresponding to this implication.

(iii) Does \( \neg \neg \neg A \supset \neg A \)? If so, give a simply-typed lambda-term corresponding to this implication.

[5 marks]

(b) (i) Give the typing rules for Peano natural numbers and their eliminator.

[2 marks]

(ii) Using the rules given above, define the addition function.

[3 marks]

(iii) Let a binary tree be either a leaf Leaf or a node Node(1,x,r) where \( l \) and \( r \) are subtrees, and \( x \) is a natural number. Give typing rules for trees corresponding to this prose description, including an eliminator.

[3 marks]

(iv) Using the rules given above, define a function size which takes a binary tree and returns the total number of nodes in the tree.

[5 marks]

(c) The zip function takes two lists, and returns a list of pairs of the elements as output. Suppose we see the following Agda type declaration for zip:

\[
\text{zip} : \forall \{ A \ B : \text{Set} \} \rightarrow \{ n : \text{Nat} \} \rightarrow \text{Vec} \ A n \rightarrow \text{Vec} \ B n \rightarrow \text{Vec} (A \times B) n
\]

Explain what this means in terms of how to call the function, and what properties the result has.

[2 marks]