

## 1996 Paper 11 Question 4

### Compiler Construction

Describe a structure that could be used to represent the abstract syntax tree of the following  $\lambda$ -expression:

$$(\lambda a. (\lambda f. f a) (\lambda n. n + 1)) 3$$

Outline the definition of an evaluator function `eval(expr, env)` that could be used to evaluate a given expression `expr` represented in this way in the context of an environment given by `env`. Pay particular attention to the treatment of bound variables and the mechanism you use for function calls. [7 marks]

Is it possible with your implementation to give it a  $\lambda$ -expression which would cause `eval` to recurse to an unlimited depth? If so, give such a  $\lambda$ -expression; if not, explain why. [3 marks]

Is it possible with your implementation to give it a  $\lambda$ -expression that causes an environment chain of unlimited length to be created during the evaluation? If so, give a  $\lambda$ -expression that would cause this; if not, explain why it is not possible. [3 marks]

In an erroneous implementation of `eval` the value of

$$(\lambda x. body)$$

is an object that does not incorporate the contextual environment, and whose call evaluates *body* in an environment derived from the environment of the call. Explain why such an implementation is wrong, giving an example  $\lambda$ -expression that would yield different results when evaluated by this implementation and yours. [7 marks]