

2007 Paper 10 Question 11

Introduction to Functional Programming

(a) Specify the types of the following SML functions:

(i) `fn f => map f` [2 marks]

(ii) `fn f => map map f` [2 marks]

(iii) `fn f => (map o map) f` [2 marks]

[Recall that the composition operator `o` has type
 $(\alpha \rightarrow \beta) * (\gamma \rightarrow \alpha) \rightarrow \gamma \rightarrow \beta.$]

(b) Let

```
datatype  $\alpha$  tree = empty | node of  $\alpha$  *  $\alpha$  tree *  $\alpha$  tree
```

be the data type of binary trees.

(i) A *bijective correspondence* between types α and β is given by a pair of functions $f : \alpha \rightarrow \beta$ and $g : \beta \rightarrow \alpha$ such that $g \circ f = \text{fn } a : \alpha \Rightarrow a$ and $f \circ g = \text{fn } b : \beta \Rightarrow b$.

Exhibit recursive SML functions `f : unit tree \rightarrow (unit tree) list` and `g : (unit tree) list \rightarrow unit tree` that establish a bijective correspondence between the types `unit tree` and `(unit tree) list`. [3 marks]

(ii) Give an alternative definition of the function `g` in terms of either `foldl` or `foldr`. [3 marks]

(iii) Define a `treefold` function of type

$$(\alpha * \alpha \text{ tree} * \beta \rightarrow \beta) \rightarrow \beta \rightarrow (\alpha \text{ tree}) \rightarrow \beta$$

and give an alternative definition of the function `f` in terms of it. [3 marks]

(c) Rigorously argue for the correctness of the following identities:

```
g o f = fn t:unit tree => t
```

```
f o g = fn l:(unit tree) list => l
```

[5 marks]