2000 Paper 5 Question 10

Foundations of Functional Programming

Give a brief account of how *four* of the following features of general programming systems can be modelled in terms of a form of un-typed functional programming where none of the mentioned facilities are provided as built-in features.

When selecting your examples and preparing your explanations, arrange that at least *one* of the four cases could be carried out using a typical polymorphically typed functional language while at least *one* would lead to type-checking problems.

(*a*) Tuples (it will be sufficient to consider just the case of pairs).

(*b*) Boolean quantities and an *if*/*then*/*else* construct.

(*c*) Lists (both empty and non-empty).

(*d*) Recursive function definitions.

(*e*) The numbers $0, 1, 2, \ldots$, with the associated operations of a zero test, addition and multiplication.

[4 marks each]

Explain the issues about type checking for all of the examples you have given.

[4 marks]