2000 Paper 1 Question 6

Foundations of Computer Science

This question has been translated from Standard ML to OCaml

Describe how OCaml lists are represented in storage. Your answer should include diagrams illustrating how the representation of `[a; b] @ [c; d]` is derived from those of the lists `[a; b]` and `[c; d]`, indicating any sharing of memory. How efficient is the evaluation of `[a; b] @ l` if the list `l` is very long? [4 marks]

What are *cyclic lists* and how can they be created in OCaml? [2 marks]

NB This question was unambiguous in Standard ML, which doesn't have OCaml's recursive value definitions.

Describe OCaml's reference types and their applications. In particular, compare mutable data structures with ordinary OCaml types. [6 marks]

Code an OCaml function that takes a mutable list and returns true if the list is cyclic, otherwise returning false. Explain why your function is correct.
[Hint: in OCaml, the physical equality test p `==` q is true if p and q refer to the same location in memory.] [8 marks]