Section B:

Question 3 I had expected better answers to at least parts (a) and (b) which are worth a combined 70%. Part (a) is standard book work and can be either the mathematical statement of induction on lists (preferred) or an equivalent wordy answer.

Part (b) requires two inductive proofs, one to show that \([\ ]\) is a right identity, and one to show that ++ is associative. Many students did not know what it meant for an operator to be associative. Even so one mark was available for showing that \([\ ]\) is a left identity, a fact which requires no induction at all!

Part (c) was slightly more taxing as it involved proving properties about infinite lists. Nevertheless, this material had been covered in both lectures and examples classes, so it was mildly disappointing that there were relatively few good answers. Where students knew what they were doing, and had revised the material presented in the lectures and examples classes, there were 20 marks available.

Part (d) Required four simple definitions from the notes for 0.5 marks each.

Part (e) Was a worked example from the lecture and example class material again.

Part (f) Was usually quite well done. We simply accept non-termination as the way to represent that a program and its input do not halt!

Question 4 By and large this question was well done.

Part (a) was usually well done though there was apparent confusion over what is meant by "partial correctness"

Part (b) This was well done; most students could show that the expression was a loop invariant.

Part (c) This was also well done.

Part (d) Given its importance in the second and third year, this material was not well done. Perhaps the issue of using a substitute lecturer for this lecture is showing in the exam results?

Part (e) There is a trick here, in that the student needs to find a recurrence formula for the recursive call tree, and not solve for the function itself!

Part (f) Surprisingly well done. The use of Big-Theta ought to be a doddle in the second year algorithms course.