2010 Paper 1 Question 5

Algorithms I

(a) Describe the basic principle of the *mergesort* algorithm. Illustrate your answer by showing the steps involved in sorting the array { 9, 3, 6, 2, 4, 1, 5 }.
[6 marks]

(b) *Insertion sort* can be considered as a mergesort where each step divides an array of size n into two arrays: one of size 1 (the element to be inserted) and one of size $(n-1)$ for array length n. By solving an appropriate recurrence relation, show that this recursive version of insertion sort has a time complexity of $O(n^2)$. Assume the time complexity for merging two arrays is $O(n)$.
[5 marks]

(c) A programmer is tasked with sorting both arrays and linked lists. For both data structures, he intends to use the mergesort algorithm.

(i) Show that the time complexity of a linked list mergesort is $O(n \log n)$. Show also that the space complexity is $O(1)$, taking care to demonstrate how this can be achieved. [6 marks]

(ii) The programmer only knows how to merge two arrays in $O(n)$ space and linked lists in $O(1)$ space, so proposes converting the arrays to linked lists before applying the mergesort algorithm to save on space. Comment on this strategy. [3 marks]