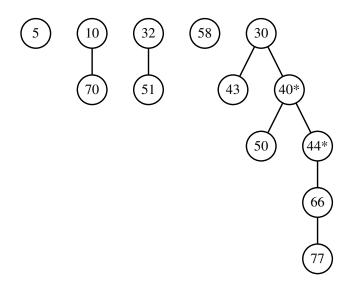## 2009 Paper 4 Question 2

Algorithms II

(a) Given the following Fibonacci heap, where nodes with an asterisk are "marked", perform `extractMin()` on it and then `decreaseKey()` on the node whose key is currently 66, bringing it down to 4. Redraw the changed heap as you go along.

You need only draw any significant intermediate states of the heap, adding any necessary explanations so that a reader can follow what you are doing and why. [5 marks]

(b) Fibonacci heaps are designed so that their trees never become excessively "wide and shallow". Why? Justify this design goal in detail and then explain how it is achieved. [5 marks]

(c) Nothing, however, stops the trees in a Fibonacci heap from growing "tall and narrow". Prove this by describing a sequence of Fibonacci heap operations that, given an integer n, produces a Fibonacci heap made of a single tree consisting of a linear chain of n nodes (in other words, each node in the tree except for the last one is the parent of exactly one node, and each node except for the first one is the child of exactly one node). [10 marks]