COMPUTER SCIENCE TRIPOS Part IB – 2014 – Paper 3

2 Algorithms II (FMS)

(*a*) Consider van Emde Boas (vEB) trees.

(*i*) On its own page for legibility, draw the smallest vEB tree storing keys 0, 3, 6, 7. The correctness of the structure and the accuracy of all fields of all nodes are important. Once done, write each of the keys under the cluster in which it is logically stored. [8 marks]

(*ii*) vEB trees store the minimum and maximum key of a subtree in the root node, but do not store the minimum key in any of the descendent clusters. Explain all the reasons why this provides a performance advantage compared with proto-vEB trees. [4 marks]

(*b*) Consider proto-vEB trees.

The following pseudocode attempts to implement a method to delete a key from a proto-vEB node. Give a clear explanation of the strategy that it uses. Fix any bugs that it may contain. Give more meaningful identifiers for the variables `c` and `s`. Give appropriate comments for the four positions marked "`COMMENT HERE`". Explain why the method returns a value and what the value means. [8 marks]

```
boolean delete(self, key)
    # HEADER COMMENT HERE (1)
    if self.u == 2:
        if self.A[key] == 0
            # COMMENT HERE (2)
            return False
    else:
        c = self.cluster[high(key)].delete(low(key))
        if c:
            # COMMENT HERE (3)
            s = self.summary.delete(high(key))
            # COMMENT HERE (4)
            return s
        else:
            return False
```