



3. Repeat Question 2 using skew merging instead of simplistic merging.

4. Given a positive integer  $n$ , describe how to generate a sequence of  $n + 1$  operations on an initially empty skew heap such that the last operation requires  $\Omega(n)$  time, or explain why no such operations are possible. Provide illustrative worked examples.

5. Depict all distinct heap-ordered binary trees over the keys 1, 2, 3, 4. You may abbreviate.

6. Provide a non-recursive variant of the recursive skew-merging algorithm of Section 23.1.3. Justify its correctness and quantify its time and space complexities, highlighting any differences from the recursive counterparts.