1.

2. A general algorithm is as follows:

   (1) Initially, let $l = 0$ and $u = N$ be lower and upper bounds for the set that contains the smallest divisor of $N$.

   (2) Loop: (divide and conquer – bsearch inspiration)

      i. Let $m = \lfloor (l + u)/2 \rfloor$.

      ii. If there is a divisor $\leq m$ adjust the upper bound $u = m$. Otherwise adjust the lower bound $l = m$.

      iii. Continue until $u - l = 1$, the final element is the smallest divisor.

   In every loop iteration the set size is cut in half. Since $N \leq 2^n$ we can have at most $n$ such iterations until the set has at most one element. The time complexity of the algorithm is thus in $O(n^4)$.