1. Suppose you are in the comparison-based model and you are given a list of \( n \) distinct numbers, \( a_1, a_2, a_3, \ldots, a_n \). Define the rank of an item \( a_j \) to be its position \( i \), in a sorted order of these numbers. So if \( \pi \) is a permutation from \( \{1, 2, \ldots, n\} \) to \( \{1, 2, \ldots, n\} \), and \( a_{\pi(1)} < a_{\pi(2)} < \cdots < a_{\pi(n)} \), then the rank of \( a_j \) is equal to the index \( i \) for which \( \pi(i) = j \).

Give a deterministic algorithm for outputting the entire set of items of rank \( 3^i \), for \( i = 0, 1, 2, 3, 4, \ldots, \lfloor \log_3 n \rfloor \). Your algorithm should use \( O(n) \) comparisons.

2. Suppose you are in the comparison-based model are you are given a list of \( n \) distinct numbers, \( a_1, a_2, a_3, \ldots, a_n \). You are also given an integer \( B \), and suppose \( B \) divides \( n \). Your job is to arbitrarily partition these \( n \) numbers into \( B \) groups \( G_1, \ldots, G_B \), so that

(a) each group \( G_i \) has \( n/B \) items, and
(b) inside of each group \( G_i \), the numbers are sorted.

First argue that if \( B = \Theta(n) \), then this can be done deterministically using \( O(n) \) comparisons. Second, show that if \( B = \Theta(n^{1/3}) \), then this requires \( \Omega(n \log n) \) comparisons for any deterministic algorithm in the comparison-based model.

3. True or False: given a list \( a_1, \ldots, a_n \), one can output a sorted list of the smallest \( n^{1/3} \) items in \( O(n) \) time.