5. Algorithm Analysis

The constructor operates in O(1) time and space because it initializes the data structures.
The initializeCandidates method operates in O(n log n) time to insert data into the priority queue, and O(n) space complexity.
The castVote method operates in O(n) time because removing an element into a priority queue needs searching, and operates in O(1) space.
The castRandomVote method operates in O(n) but it requires O(n) to store the random votes temporarily since it processes before the votes are cast.
The rigElection method operates in O(n log n) time to rebuild the priority queued to "rig" the election and O(n) space.
The getTopKCandidates and auditElection methods run in O(n log n) time because they sort parameter data, and run in O(n) space since it needed the space to store the sorted results.
The setTotalVotes method has O(1) time and space.

So, the overall time complexity of this code is O(log n) and the space complexity is O(n).