1. Let n denote the length of words, m denote the length of puzzles, L denote the total input length, and s denote the length of puzzles[i]. Use bit packing to represent a set of chars, and for each puzzle, enumerate its subsets in $O(2^s)$. $O(L + m \cdot 2^s)$.
2. Compute the result for each possible subset of 2^Σ with size s, using Möbius transform for set power series. $O(|\Sigma| \sum_{i=0}^{s} \binom{|\Sigma|}{i})$.

Submission Detail

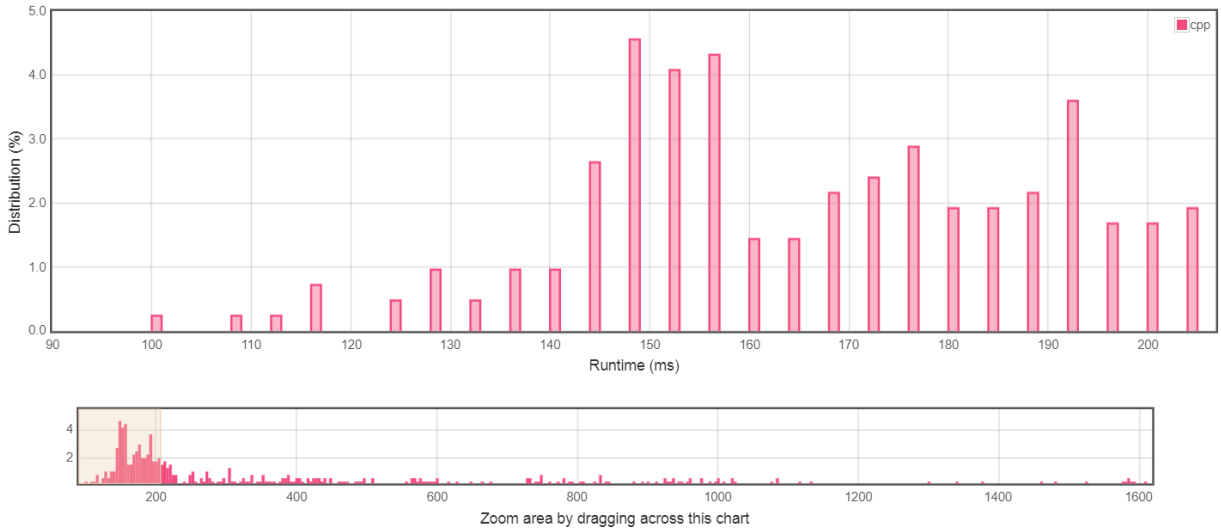
10 / 10 test cases passed.

Status: Accepted

Runtime: **44 ms**
Memory Usage: **71 MB**

Submitted: **0 minutes ago**

Accepted Solutions Runtime Distribution



Runtime: 44 ms, faster than 100.00% of C++ online submissions for Number of Valid Words for Each Puzzle.

Memory Usage: 71 MB, less than 100.00% of C++ online submissions for Number of Valid Words for Each Puzzle.

References