Group Anagrams

For this problem we need to group words that are anagrams of each other.

Key Idea:

Two strings are anagrams if they have the same character counts.

Common approaches:

1. Sort characters in each string → use as key
2. Use character frequency (26-length vector) → use as key.

Approach 1: Sorting Key:

Algorithm:

- For each string:
  1. Sort the characters → forms a canonical key
  2. Store string in a hash map grouped by this key.
- Return grouped values.

Approach 2: Frequency Count Key (Optimized):

- Create key using counts of 26 letters:
  - Example: "eat" → 1#0#0#...#1#...#1 (for a, e, t)
- Avoid sorting ($O(k \log k)$) → $O(k)$ per string.

Complexity:

- Approach 1:
  - Time: $O(N * K \log k)$ (K = max word length)
  - Space: $O(N * K)$
- Approach 2:
  - Time: $O(N * K)$
  - Space: $O(N * 26)$