You have n tiles, where each tile has one letter tiles[i] printed on it. Return the number of possible non-empty sequences of letters you can make using the letters printed on those tiles

CODE:

import java.util.HashMap;

import java.util.Map;

import java.util.Scanner;

public class LetterSequences {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the number of tiles: ");

int n = scanner.nextInt();

char[] tiles = new char[n];

System.out.println("Enter the letters on the tiles:");

for (int i = 0; i < n; i++) {

tiles[i] = scanner.next().charAt(0);

}

int sequences = countSequences(tiles);

System.out.println("Number of possible non-empty sequences: " + sequences);

scanner.close();

}

public static int countSequences(char[] tiles) {

Map<String, Integer> memo = new HashMap<>();

int[] freq = new int[26];

for (char tile : tiles) {

freq[tile - 'A']++;

}

return backtrack(freq, memo);

}

private static int backtrack(int[] freq, Map<String, Integer> memo) {

String key = getKey(freq);

if (memo.containsKey(key)) {

return memo.get(key);

}

int total = 0;

for (int i = 0; i < 26; i++) {

if (freq[i] > 0) {

freq[i]--;

total += 1 + backtrack(freq, memo);

freq[i]++;

}

}

memo.put(key, total);

return total;

}

private static String getKey(int[] freq) {

StringBuilder sb = new StringBuilder();

for (int f : freq) {

sb.append(f).append("#");

}

return sb.toString();

}

}

OUTPUT:

C:\javap>javac LetterSequences.java

C:\javap>java LetterSequences

Enter the number of tiles: 4

Enter the letters on the tiles:

A B C D

Number of possible non-empty sequences: 64

