Codeforces :: Daily Practice

[+] E1. Beautiful Patterns (Easy Version)

Given a string representing a mosaic, the task is to compute the expected beauty value. The beauty is defined based on the number of palindromic segments formed after random partitioning. This version has smaller constraints and focuses on understanding the combinatorial structure and probability calculation using prefix properties.

-> https://codeforces.com/contest/2175/problem/E1

[+] E2. Beautiful Patterns (Hard Version)

This is the harder version of E1 with significantly larger constraints. The core idea remains the same, but the solution requires optimized algorithms and careful handling of large input sizes. Advanced prefix techniques and efficient counting are necessary to pass within time limits.

-> https://codeforces.com/contest/2175/problem/E2

[+] F. Secret Message

You are given a weighted graph and must select exactly n-1 edges such that they do not form a tree. The goal is to minimize the total weight. This problem requires deep understanding of graph theory, especially spanning trees, cycles, and complementary edge selection strategies.

-> https://codeforces.com/contest/2175/problem/F

[+] E. Remove at the lowest cost

Given an array, you can perform removal operations with different costs, including some zero-cost operations. The task is to remove all elements with the minimum total cost. This problem is solved using greedy strategies or dynamic programming, depending on the interpretation of operations and constraints.

-> https://codeforces.com/contest/2176/problem/E

[+] F. Omega Numbers

For a given array, define omega(x) as the number of prime factors of x. The task is to compute the sum of omega(ai * aj)^k over all pairs i < j. This problem combines number theory with combinatorics and requires fast factorization and precomputation techniques.

-> https://codeforces.com/contest/2176/problem/F