Wiring

Subtask 1

There are many different Dynamic Programming (DP) solutions with $O(n^2)$ or $O(n^3)$ running time for this subtask. The simplest one is to define $dp_{i,j}$ as the minimum cost needed for wiring the first i red points and the first j blue points. Update is like:

$$dp_{i,j} = \min(dp_{i-1,j}, dp_{i,j-1}, dp_{i-1,j-1}) + |red_i - blue_j|$$

Subtask 2

This subtask is to find the pattern of wiring. A simple solution to this subtask is to calculate:

$$\sum_{0}^{n-1}(red_{n-1} - red_i) + \sum_{0}^{m-1}(blue_i - blue_0) + \max(n, m) \times (blue[0] - red[n-1])$$

Subtask 3

Consider the consecutive clusters of points with the same color. The idea is that each wire will have endpoints in two consecutive clusters, so the $O(n^2)$ solutions could be optimized to $O(n \times MaxBlockSize)$.

Subtask 4

This subtask could be solved by a greedy algorithm that divides each cluster into two halves and connects the left half to the left cluster and the right half to the right cluster. The middle point of a cluster with an odd number of points should be considered separately.

Optimal solution

There is $O(n + m)$ DP solution: let dp_i be the minimum total distance of a valid wiring scheme for the set of point i and all points to the left of it. This could be updated with an $O(1)$ amortized time complexity.