

2002 Sun Microsystems and TopCoder Collegiate Challenge – Problem Analysis

Nested:

This problem is a clear candidate for dynamic programming. Deleting symbols and inserting symbols are equivalent operations. Both milhouse and po realized this and tried to exploit the dualism by ignoring insertions and searching the possible ways to delete symbols to form a balanced string. However, there are far too many substrings to explore in eight seconds, even using a shortcut to eliminate balanced substrings (as milhouse implemented in his solution).

Building a 2D table containing the best cost for all substrings determines the solution in polynomial time. The cost balancing a single symbol is $\min(\text{delCost}, \text{insCost})$. The cost of balancing $\text{data.substring}(a,b)$ equals the minimum of $\text{cost}(\text{data.substring}(a,x)) + \text{cost}(\text{data.substring}(x,b))$, over all $a < x < b$, and of $\text{cost}(\text{data.substring}(a+1,b-1))$ if the $\text{data}[a]$ and $\text{data}[b]$ match. This approach was used by both ante and malpt. However, ante's solution breaks on an empty input, leaving malpt with the only working solution.