

15-351 / 15-650 / 02-613: Exercises 2c

1. Given an $N \times 3$ matrix, you should fill in each cell with one color from red, green and blue, such that every neighbouring cells (up, down, left, right) do not share the same color. For example, for every (i, j) , $matrix[i - 1, j]$ (up), $matrix[i + 1, j]$ (down), $matrix[i, j - 1]$ (left), $matrix[i, j + 1]$ (right) should not equal to $matrix[i, j]$, where $matrix[x, y]$ represents the color filled in cell (x, y) . Note that all cells need to be colored.

Please derive a dynamic programming algorithm to find the maximum number of ways to color the matrix. In your algorithm, the input is N and the output is the maximum number of ways to color the matrix.

2. Given the string “IWANTADISCOVERYURGENTLY”, you want to add spaces to get to the phrase “I WANT A DISCOVERY URGENTLY”. Unfortunately, there might be multiple ways to split the string (“I WANT A DISCO VERY URGENTLY”), so to formalize the problem, we assign every word w a score $\text{score}(w)$ by how common it is, and seek to maximize the summed word scores, where the score of a nonsense word (“NTLY”) is much lower than the score of a super rare word (“ANTA”), which is itself lower than the score of a highly likely common word (“WANT”).

More formally, suppose you are given a string $a = a_1a_2 \dots a_n$ without spaces that you want to break up into an **unknown** number k of words $w_1, \dots, w_k$ such that you are maximizing the summed word score $\sum_i \text{score}(w_i)$. Give a dynamic programming algorithm to solve this problem in $O(n^3)$ time at most, assuming that the $\text{score}(w)$ function call takes $O(1)$ time.

3. Let $A = a_1, a_2, \dots, a_N$, $B = b_1, b_2, \dots, b_N$, and $C = c_1, c_2, \dots, c_N$ be three sequences of length N . The longest common subsequence is the longest sequence $S = s_1, \dots, s_K$ such that S is a subsequence of A , B , and C . Note that a subsequence is defined as any sequence that can be formed by deleting one or more characters from a sequence. For example, “abd” and “be” are valid subsequences of the sequence “abcde”. Provide a dynamic programming algorithm that, given sequences A , B , and C , finds the length of their longest common subsequence. Briefly explain why your algorithm is correct and provide the runtime.