Text Justification

You are given a sequence of words, and a limit on the number of characters that can be put in one line (i.e., the line width). Put line breaks in the given sequence such that the lines are printed neatly. Assume that the length of each word is smaller than the line width.

Input: Given array of $n + 1$ words $w[0 : n]$.

Measure of badness (ugliness): Suppose we are considering a line ℓ containing the words $w[i]$ through $w[j]$. Define the badness(ℓ) for the line of words $\ell := w[i : j + 1]$ to be:

$$badness(\ell) = \begin{cases} (\text{page-width} - \text{total-length}(\ell))^3, & \text{if total-length}(\ell) \leq \text{page-width}\,; \\ +\infty, & \text{otherwise}\,. \end{cases}$$

Goal: Split words into lines $\ell_1 = w[0 : i_1]$, $\ell_2 = w[i_1 : i_2]$, etc. to minimize $\sum_\ell badness(\ell)$.

Question 1: Write a dynamic programming algorithm to solve this problem. You should first define the subproblems and recursive relation.

Subproblem: $Just(i)$: the sum of badness measures of the remaining lines given $w[i :]$
Guess: word starting of the second line
Number of choices: at most $n - i$ which is $O(n)$

Recursive Relation:

$$Just(i) = \begin{cases} 0, & \text{if } i = n \\ \min_{i+1 \leq j \leq n} \Big(Just(j) + badness(w[i : j + 1]) \Big), & \text{otherwise.} \end{cases}$$

Original Problem: $Just(0)$

Question 2: Determine the number of subproblems that you have and analyze the running time of this algorithm.

Number of distinct subproblems is $O(n)$
Computation per subproblem is $O(n)$
Total running time $O(n^2)$