

2016 Spring CS300 Programming Assignment #2

(Due: June 5 PM 11:59)

Modern computing is human-centric. Its focus has shifted significantly towards humans – the creators, users, and subjects of computing – from the hardware and software technologies. This paradigm shift, as demonstrated by the growing interest in social computing with big data, presents opportunities for a deeper

Suppose we would like to neatly typeset a text. The input is a sequence of n words of lengths l_1, l_2, \dots, l_n (measured in the number of fixed-size characters they take.). Each line can hold at most P characters, the text is left-aligned, and words cannot be split between lines. If a line contains words from i to j (inclusive) then the number of spaces at the end of the line is

$$s = P - (j - i) - \sum_{k=i}^j l_k.$$

We would like to typeset the text so as to avoid large white spaces at the end of lines. Your task is to minimize the sum over all lines of the number of white spaces at the end of the line using dynamic programming in $O(n^2)$ time. (Hint: Let $A[j]$ denote the optimal ‘cost’ (that is, the sum of the square of number of trailing white space characters over all lines) one may achieve by typesetting only the words $1 \dots j$ (ignoring the remaining words).)

Input

Input starts with a line containing two integers n , and P ($1 \leq n \leq 1,000$; $1 \leq P \leq 100$). P is always bigger than maximum length among l_1, l_2, \dots, l_n . Next line contains a sequence of n words of lengths l_1, l_2, \dots, l_n .

Output

Display the minimized sum over all lines of the square of the number of white spaces at the end of the line.

Sample Input	Output for Sample Input
5 5 3 2 1 1 2	4
6 10 3 4 5 3 6 7	10