CS 124 Problem Set 5

Logged in as siddharthsingh@fas.harvard.edu

Due: Wednesday, March 25, 2015 11:59 pm EDT (**deadline passed**)

Problems

Break the String

A programmer would like to insert L line breaks into a string after
b1, ..., b_L characters.
Since he is working on an old computer, which must rerender after each line
break, it takes n milliseconds to break a string of n characters
(line breaks are not counted in the number of characters) into two pieces.

Consequently, the amount of time required is affected by the order
in which the line breaks are added. For example, given the string
"needlinebreaks", it takes 14+10=24 milliseconds to first break after
"need" and then after "line", whereas making those two line breaks
in the opposite order only requires 14+8=22 milliseconds.

Your goal is to determine the minimum amount of time, in milliseconds,
required for the programmer to make all the line breaks.

CONSTRAINTS
For test cases worth a total of 0 points:
There is a single test case (test 1) that is worth 0 points.
The input is simply the example in the spec, below ("SAMPLE INPUT").
This test serves to verify that the server is assessing your code correctly, in the
event that all of the other test cases are reported as Incorrect Output.

0 ≤ [string length] ≤ 1000000

For test cases worth a total of 30 points:
0 ≤ L ≤ 10

For test cases worth a total of 60 points:
0 ≤ L ≤ 100

For test cases worth a total of 10 points:
0 ≤ L ≤ 1000

TIME LIMIT
100 ms per test cases in first and second group. (2x for Java, 10x for Python)
200 ms per test cases in third group.

INPUT FORMAT
First a line containing L (the number of line breaks) and l (the length of the string).
Next a line containing b1 through b_L, the points
at which line breaks are to be made, with 0 < b1 < ... < b_L < l.

```
OUTPUT FORMAT
Print a single integer representing the minimum number of
milliseconds to make all of the line breaks.

SAMPLE INPUT
2 14
4 8

SAMPLE OUTPUT
22

DETAILS
This input corresponds to the example of "needlinebreaks" given in the problem statement.
```

Based on the "Ultra Cool Programming Contest Control Centre" v1.7b by Sonny Chan
Modified for CS 124 by Neal Wu, with design help from Martin Camacho