

# 决策单调性

# Noi2009诗人小G

- 小 G 是一个出色的诗人，经常作诗自娱自乐。但是，他一直被一件事情所困扰，那就是诗的排版问题。
- 一首诗包含了若干个句子，对于一些连续的短句，可以将它们用空格隔开并放在一行中，注意一行中可以放的句子数目是没有限制的。小 G 给每首诗定义了一个行标准长度（行的长度为一行中符号的总个数），他希望排版后每行的长度都和行标准长度相差不远。显然排版时，不应改变原有的句子顺序，并且小 G 不允许把一个句子分在两行或者更多的行内。在满足上面两个条件的情况下，小 G 对于排版中的每行定义了一个不协调度，为这行的实际长度与行标准长度差值绝对值的  $P$  次方，而一个排版的不协调度为所有行不协调度的总和。
- 小 G 最近又作了几首诗，现在请你对这首诗进行排版，使得排版后的诗尽量协调（即不协调度尽量小），并把排版的结果告诉他。

输入文件中的第一行为一个整数  $T$ ，表示诗的数量。

接下来为  $TT$  首诗，这里一首诗即为一组测试数据。每组测试数据中的第一行为三个由空格分隔的正整数  $N, L, P$ ，其中： $N$  表示这首诗句子的数目， $L$  表示这首诗的行标准长度， $P$  的含义见问题描述。

从第二行开始，每行为一个句子，句子由英文字母、数字、标点符号等符号组成（ASCII码33 ~ 127，但不包含'-'）。

简单的DP：

$$F[i] = \min_{j=0}^{i-1} \{F[j] + |\text{SumL}[i] - \text{SumL}[j]| + (i - j - 1) * L[P]\}$$

# 加个贪心：

可以想到每一行的诗句的总长度不会超过 $2 * l$ ，很容易证明，因为长度超过 $2 * l$ 以后总可以拆成2行使决策更优。

换句话说就是 $s[i] - s[j] < 2 * l$ 。在倒序枚举 $j$ 时若不满足循环就可以跳出了。

- 那么我们考虑如何利用决策单调性来优化这道题目
- 可以看成  $F[i] = \min(f[j] + w[i, j])$
- 首先，对于一维 dp，设  $g[i]$  为状态  $i$  的最优决策点，它的决策单调性的显著特征自然是： $g[i-1] \leq g[i]$ ，那么我们考虑利用这种决策单调性来做题。
- 算法描述：
- 维护一个栈，保存每一个决策点能决策的区间，二分查找当前点被维护的区间，求出当前点之后的以当前点为决策点入栈维护区间。

# 证明：

首先我们回到问题，我们已经选出了一些句子分好了行，在当前行我们还未换行。

不考虑下一行，那么如果还有句子加上当前的句子的长度小于标准长度，则一定要选，若长度之和大于标准长度，则只需要考虑选与不选。

所以，对于当前行的决策，我们只会浮动在两个左右的决策点之间，因为  $w$  函数的指数性递增决定了我们选择的单调性。

所以对于越往后的区间，它的最优决策点就越往后，因为  $w$  函数过大会造成  $dp$  的变劣。







