T1

题目要求实现一个拼写检查系统，给定一个词表和一个待检查的单词，判断这个单词是否在词表中，如果不在词表中，则输出一个相似的单词。拼写相似性的判断基于以下三种情况：

1. 漏写了一个字母。
2. 多写了一个字母。
3. 将某处的一个字母写成了另一个字母。

可以直接判定这个问题，漏写或多写只要看一个两个字符串是否长度差1，且一者为另一者的子序列，可以直接每次找第一个相同的暴力匹配。

如果是错写，可以看下前缀最多有多少个一样，后缀最多有多少个一样，如果前 i 个和后 $n-i-1$ 个可以匹配说明可以看作错写。

对于这个问题，我们也可以使用动态规划来计算两个字符串之间的编辑距离，然后找到编辑距离最小的单词作为相似的单词。如果编辑距离为0，则表示两个单词相同；如果编辑距离为1，则表示存在一种情况下的拼写错误；如果编辑距离大于1，则表示没有找到相似的单词。

下面是算法的步骤：

1. 读取输入数据：待检查的单词和词表中的单词。
2. 遍历词表中的每个单词，计算其与待检查单词的编辑距离。
3. 找到编辑距离最小的单词，并将其作为相似的单词输出。
4. 如果编辑距离为0，则表示待检查的单词在词表中出现，直接输出该单词。
5. 如果找不到编辑距离为1的单词，则输出"NOANSWER"。

这样，我们就可以根据编辑距离来判断单词的相似性，并输出相应的结果。

算法的时间复杂度取决于词表的大小和单词的长度，为$O(N \times |S|)$，其中N为词表中单词的数目，S为单词的长度。

T2

30 分

显然每个后缀最多操作一次，于是暴力枚举即可，时间复杂度 $\mathcal{O}(2^n)$。

60 分

可能存在一些复杂度较大的神奇做法。

正解

签到题啊，假设令 $S_0=0$，那么答案就是 $\max(\sum_{i=1}^{n}[S_i \neq S_{i-1}]-1,0)$。为什么呢？

注意到相邻相同的字母是可以缩在一起的，那么问题变成了 `01010101...` 这种交替串将其变成不降串最小操作次数，这是非常简单的，从后往前依次翻转即可。

例如 `01010`，先变成 `01011`，再变成 `01000`，最后变成 `01111`。

T3

30 分

暴力即可。

60 分

暴力跳肯定是对的，因为集合 S 中最多就 $q-1000$ 个数，所以复杂度最坏 $\mathcal{O}(1000q)$。

正解

实际上暴力就是对的（加上记忆化）！

考虑只有加数没有删除数字，那么对于一个固定的 k，这个 t 肯定是不降的，于是每次记忆化即可。

考虑这个算法的最坏复杂度，最坏情况下肯定是先加入 $1,2,\ldots,q$，然后询问 $1,2,\ldots q$，这个复杂度是调和级数级别的，所以复杂度就是 $\mathcal{O}(q\log q)$。

T4

20 分

暴力搜索即可。

40 分

需要挖掘一些性质，考虑到最终留下来的序列是原序列的一个子序列。

那么一个合法的子序列需要满足什么性质。

假设这个子序列是 $A_{i_1},A_{i_2},\ldots,A_{i_k}(i_1<i_2<\ldots<i_k)$，该子序列合法的充要条件是：

- $\forall 1\le j<i_1$，$A_j>A_{i_1}$。
- $\forall i_k<j\le n$，$A_j>A_{i_k}$。
- $\forall i_p<j<i_{p+1}$，$A_j>A_{i_p}$或者 $A_j>A_{i_{p+1}}$。

这个条件是显然的，因为如果区间 $[i_p+1,i_{p+1}-1]$ 的最小值比两边留下来的数都小，那么这个数肯定删除不了。

知道这个就好搞了，暴力搜索枚举即可，复杂度是指数级的。

60 分

注意到这个是 dp 的，设 dp_i 表示选到了 i（强制选 i）的最长合法的子序列长度。

转移是 easy 的，枚举上一次的转移点，合法条件暴力 check 即可，时间复杂度是三方的。

80 分

每次判断合法没必要暴力 check，知道这个区间的最小值就行了，于是从后往前维护最小值即可。

时间复杂度是两方的。

正解

考虑优化，设 pos_i 表示 a_i 左边第一个比它小的数的位置。

枚举上一次的转移点 j。

- $j \in [pos_i + 1, i - 1]$，注意到这样的转移点一定是合法的，前缀和优化即可。
- $j \le pos_i$，如果 $a_j > a_i$，那么这个转移点显然不合法，否则需要满足 $\min_{k=j}^{i} a_k = a_j$，即所有后缀最小值的位置都是合法的转移点，维护一个 $g_k = f_k + f_{l_k} + f_{l_{l_k}} + \dots$ 即可。

时间复杂度 $\mathcal{O}(n)$。