

NOIP 2019 模拟赛 Contest 7

diamond_duke

题目名称	字符串	序列	交换
可执行文件名	str	seq	swap
输入文件名	标准输入	标准输入	标准输入
输出文件名	标准输出	标准输出	标准输出
时间限制	1s	1s	1s
内存限制	512MB	512MB	512MB
子任务个数	4	5	6
题目类型	传统型	传统型	传统型

请注意： 评测时开启 O2 优化和 C++11 编译选项，栈空间限制同空间限制。

1 字符串

1.1 Problem Statement

小 D 正在研究字符串。

小 D 已经对字符串的最长公共子序列问题非常熟悉了。如果你并不知道这是什么，下面给出了这个问题的定义：

对于一个字符串 $S = S_1S_2\cdots S_n$ ，定义他的一个**子序列**是任意满足 $1 \leq i_1 < i_2 < \cdots < i_k \leq n$ 的字符串 $S_{i_1}S_{i_2}\cdots S_{i_k}$ 。

对于两个字符串 S, T ，我们定义他们的**最长公共子序列**为一个最长的字符串，使得它既是 S 的子序列，又是 T 的子序列。

小 D 觉得最长公共子序列问题没有什么挑战，因此他决定研究这个问题的反方向：**最短公共非子序列**。其中，字符串 S 的**非子序列**指不是 S 的子序列的字符串。

对于两个字符串 S, T ，我们定义他们的**最短公共非子序列**为一个最短的字符串，使得它既是 S 的非子序列，又是 T 的非子序列。

为了降低难度，小 D 决定只研究两个 0/1 字符串的最短公共非子序列。小 D 又觉得这个问题过于简单了，因此他还要求出这两个字符串**字典序最小**的最短公共非子序列。其中，我们认为字符串 $A = A_1A_2\cdots A_k$ **字典序**小于字符串 $B = B_1B_2\cdots B_k$ ，当且仅当存在某个 $1 \leq i \leq k$ ，使得 $A_i < B_i$ 且 $\forall 1 \leq j < i$ ，有 $A_j = B_j$ 。

这下小 D 不会了，请你帮帮他。

1.2 Input Format

从标准输入读入数据。

第一行两个正整数 n, m ，表示两个字符串 S, T 的长度。

第二行一个长度为 n 的，仅由 0/1 组成的字符串 S ，表示第一个字符串。

第三行一个长度为 m 的，仅由 0/1 组成的字符串 T ，表示第二个字符串。

1.3 Output Format

向标准输出输出答案。

输出一行一个字符串，表示字典序最小的，最短公共非子序列。

1.4 Sample 1

1.4.1 Input

```
4 7
0101
1100001
```

1.4.2 Output

```
0010
```

1.5 Sample 2

见下发文件 `str/str2.in` 与 `str/str2.ans`。

1.6 Sample 3

见下发文件 `str/str3.in` 与 `str/str3.ans`。

1.7 Constraints

对于所有测试数据， $1 \leq n, m \leq 4000$ 。

- 子任务 1 (15 分): $n, m \leq 10$;
- 子任务 2 (20 分): $m \leq 10$;
- 子任务 3 (25 分): $n, m \leq 500$;
- 子任务 4 (40 分): 无特殊限制。

2 序列

2.1 Problem Statement

小 D 正在研究序列。

如果序列中的数字太大，那么小 D 研究起来会非常困难。因此，小 D 将只研究所有**数字之和不超过 N 的正整数数列**。

对于一个序列 a_1, a_2, \dots, a_n 满足 $\sum_{i=1}^n a_i \leq N$ ，小 D 定义它的**频率**为序列 b_1, b_2, \dots, b_N ，其中 b_i 表示 i 在序列 $\{a\}$ 中的出现次数。

小 D 定义函数 $f(x)$ 表示 x 在二进制下 1 的个数。小 D 定义一个序列的**权值**为频率的函数值之和，即 $\sum_{i=1}^N f(b_i)$ 。

小 D 想知道，所有和不超过 N 的序列的权值最大值是多少。

这下小 D 不会了，请你帮帮他。

2.2 Input Format

从标准输入读入数据。

第一行一个正整数 T ，表示测试数据组数。

接下来 T 行，每行一个正整数 N 表示一组数据中，对和的限制。

2.3 Output Format

向标准输出输出答案。

输出 T 行，每行一个正整数，表示权值最大值。

2.4 Sample 1

2.4.1 Input

1
7

2.4.2 Output

3

2.4.3 Explanation

一组最优解是 $a = [1, 1, 1, 4]$ ，此时 $b = [3, 0, 0, 1]$ ，因此权值为 3。

2.5 Sample 2

见下发文件 `seq/seq2.in` 与 `seq/seq2.ans`。

2.6 Sample 3

见下发文件 `seq/seq3.in` 与 `seq/seq3.ans`。

2.7 Constraints

对于所有测试数据, $1 \leq T \leq 10^3$, $1 \leq N \leq 10^{18}$ 。

- 子任务 1 (15 分): $N \leq 10$;
- 子任务 2 (20 分): $N \leq 30$;
- 子任务 3 (25 分): $N \leq 10^9$;
- 子任务 4 (20 分): $N \leq 10^{14}$ 。
- 子任务 5 (20 分): 无特殊限制。

3 交换

3.1 Problem Statement

小 D 正在研究交换。

小 D 认为一个整数序列是**好的**，当且仅当它先（不严格）上升，后（不严格）下降。形式化地，我们认为序列 a_1, a_2, \dots, a_n 是好的，当且仅当存在某个 $k \in [1, n]$ ，使得对于任意 $1 \leq i < k$ ，有 $a_i \leq a_{i+1}$ ；且对于任意 $k \leq i < n$ ，有 $a_i \geq a_{i+1}$ 。

小 D 得到了一个长度为 n 的序列 a_1, a_2, \dots, a_n ，他让这个序列变成好的。

小 D 每次可以交换相邻的两个元素。因为交换很累，所以小 D 想知道，他最少需要交换多少次。

这下小 D 不会了，请你帮帮他。

3.2 Input Format

从标准输入读入数据。

第一行一个正整数 n ，表示序列的长度。

第二行 n 个空格隔开的整数 a_1, a_2, \dots, a_n ，表示初始的序列。

3.3 Output Format

向标准输出输出答案。

输出一行一个整数，表示最小交换次数。

3.4 Sample 1

3.4.1 Input

```
6
6 5 1 3 2 4
```

3.4.2 Output

```
4
```

3.4.3 Explanation

一组最优解如下：

- 交换 a_5, a_6 ，得到 $a = [6, 5, 1, 3, 4, 2]$ ；
- 交换 a_4, a_5 ，得到 $a = [6, 5, 1, 4, 3, 2]$ ；

- 交换 a_2, a_3 , 得到 $a = [6, 1, 5, 4, 3, 2]$;
- 交换 a_1, a_2 , 得到 $a = [1, 6, 5, 4, 3, 2]$ 。

3.5 Sample 2

见下发文件 `swap/swap2.in` 与 `swap/swap2.ans`。

3.6 Sample 3

见下发文件 `swap/swap3.in` 与 `swap/swap3.ans`。

3.7 Constraints

对于所有测试数据, $1 \leq n \leq 3 \times 10^5$, $1 \leq a_i \leq n$ 。

- 子任务 1 (15 分): $n \leq 10$;
- 子任务 2 (20 分): $n \leq 500$;
- 子任务 3 (15 分): $n \leq 5000$;
- 子任务 4 (15 分): $n \leq 10^5$;
- 子任务 5 (20 分): $\{a\}$ 是一个 $[1, n]$ 的排列;
- 子任务 6 (15 分): 无特殊限制。