

T1. 字符串 (a.cpp 1S 512M)

1. 题面描述:

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 $S = S_1 S_2 \cdots S_n$ ，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 $A = A_1 A_2 \cdots A_k$ **字典序**小于字符串 $B = B_1 B_2 \cdots B_k$ ，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。

2. 输入格式:

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。

3. 输出格式:

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。

4. 样例输入:

```
4 7
0101
1100001
```

5. 样例输出:

```
0010
```

6. 数据范围:

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。

T2. 排列 (b.cpp 1S 512M)

1. 题面描述

给定一个长度为 n 的全排列 $[a_1, a_2, \dots, a_n]$ ，称这个数组是一个 m 排列，当且仅当存在 $1 \leq l \leq r \leq n$ ， $m = r - l + 1$ ，使得 $[a_l, a_{l+1}, \dots, a_r]$ 恰好是 1 到 m 的一个排列。

例如数组 $[4, 5, 1, 3, 2, 6]$ ，是一个 1 排列 ($[1]$)、3 排列 ($[1, 3, 2]$)、5 排列 ($[4, 5, 1, 3, 2]$)、6 排列 ($[4, 5, 1, 3, 2, 6]$)。

显然长度为 n 的全排列，必然是 1 排列和 n 排列。

现在的问题是，给出一个数组，对于所有的 $1 \leq m \leq n$ ，请问这个数组是否是一个 m 排列。

2. 输入格式

输入数据第一行是一个整数 n ，第二行包含 n 个整数。

保证这 n 个整数是 $1 \sim n$ 的全排列。

3. 输出格式

输出数据包含一个 n 位的 01 字符串，对于第 i 位 ans_i ：

1. 若 $ans_i = 0$ 表示这不是一个 i 排列；
2. 若 $ans_i = 1$ 表示这是一个 i 排列。

4. 样例输入

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

5. 样例输出

```
101011
```

6. 数据范围

30% 的数据满足 $1 \leq n \leq 5$ ；

60% 的数据满足 $1 \leq n \leq 10$ ；

100% 的数据满足 $1 \leq n \leq 10000$ 。

T3. 序列 (c.cpp 1S 512M)

1. 题面描述

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. 输入格式

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。

3. 输出格式

输出 T 行，每行一个正整数，表示权值最大值。

4. 输入样例

1
7

5. 输出样例

3

6. 样例解释

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

7. 数据范围

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。

T4. 交换 (d.cpp 1S 512M)

1. 题面描述

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 ，他想要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。

2. 输入格式

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. 输出格式

输出一行一个整数，表示最小交换次数。

4. 输入样例

```
6
651324
```

5. 输出样例

```
4
```

6. 样例解释

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 $a = [1, 6, 5, 4, 3, 2]$ 。

7. 数据范围

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。