

2020普转提七连测day3

A.aria的礼物

题面描述

Aria 的生日快到了, Jane 为她准备了一个用密码箱包装起来的礼物。要解开这个密码箱需要解决如下一个谜题。

Jane 给出了目前有一个长度为 n 的全由小写字母构成的字符串。然后改变其中最少数量的字符数 (改变的方法是可以将任意一位小写字符替换成另一个小写字符), 使得前 k 个字符和后 k 个字符相同。这个最少改变的字符数就是这个密码箱的密码。

Jane 给出了 n, k , 以及这个字符串。Aria 显然懵了, 于是她找到了你求助, 希望你能给出答案。

输入格式

第一行包括两个正整数 n, k 。

第二行一个长度为 n 个小写字符串。

输出格式

一行一个非负整数, 表示最少修改的字符数量。

样例

输入1

```
10 7
```

```
aabaabaabc
```

输出1

```
1
```

输入2

```
10 7
```

```
ababababab
```

输出2

```
4
```

样例解释

样例1: aabaabaaba

样例2: abaabaabaa

数据规模

对于 50% 的数据: $1 \leq k \leq n \leq 20$, 且字符仅有 a, b 构成。

对于 100% 的数据: $1 \leq k \leq n \leq 10^5$ 。

额外限制: 其中有 50% 的数据点满足: $1 \leq k \leq \lfloor \frac{n}{2} \rfloor$, 且保证均匀分布。

时间限制: 1S

空间限制: 256MB

B.神奇位运算

题面描述

CLB 有你 n 个长为 m 的 01 串，和一台神奇的机器。

这台机器需要由两个长度为 m 的 01 串作为输入，然后可以得到一个长度为 m 的 01 串作为输出。

输出的每一位是两个 01 串的对应位的 *and*, *or*, *xor* 三种运算中的一种的结果。你需要确定机器在每一位上究竟会做何种运算。

CLB 可以用这 n 个 01 串的任意一对来测试这台机器，并且可以测试无数次，但是 CLB 可能依然不能确定某一位机器究竟是做的何种运算。

现在 CLB 想在这 n 个 01 串的基础上再添加 k 个长度为 m 的 01 串（这 k 个 01 串可以是任意的，每一位是由自己构造的）。CLB 想知道最小的 k ，使得用这 $n + k$ 的 01 串的任意一对来测试这台机器，并且可以测试无数次，最终他能知道每一位机器做的何种运算。

CLB 把这个问题抛给了你，希望你给出最小的 k 。

输入格式

本题采用多组数据测试，一行包括一个正整数 T ，表示数据的组数。

对于每一组数据：

第一行包括两个正整数 n, m 。

之后 n 行包括 n 个长度为 m 的 01 串。

输出格式

对于每一组数据，都包括一行非负整数，表示最少的 k （即需要添加的字符串）。

样例

输入

2

3 2

11

00

10

2 8

00000000

00001111

输出

1

2

样例解释

1.所有可能的答案是 11 或 01。

2.可能的一组答案是 11110000, 11111111。

数据规模

对于 30% 的数据, $n, m \leq 5$ 。

对于 60% 的数据, $n, m \leq 20$ 。

对于 100% 的数据, $n, m \leq 500, T \leq 10$ 。

时间限制：1S

空间限制：256MB

C.终焉之数

题面描述

CLB 有一个长度为 n 的数列 a 。

但是他觉得这个数列不够酷，于是他希望能在这个数列上面加加减减，最后使得新数列的两两最大公约数为1，且最小化改变的值之和。

换言之，他希望你求得一个等长的数列 b 使得 b 数列中的两两最大公约数为 1 的同时，最小化

$$\sum_{i=1}^n |a_i - b_i|.$$

输入格式

第一行包括一个正整数 n 。

第二行包括 n 个正整数 a_i 。

输出格式

一行一个非负整数，表示最小化的 $\sum_{i=1}^n |a_i - b_i|$ 的 b 数列，如果有多个数列 b 满足要求，输出任意一个即可。

样例

输入

5

1 6 4 2 8

输出

1 5 3 1 8

数据规模

对于 30% 的数据: $1 \leq n \leq 5$ 。

对于 50% 的数据: $1 \leq n \leq 10$ 。

对于 70% 的数据: $1 \leq n \leq 30$ 。

对于 100% 的数据: $1 \leq n \leq 100, 1 \leq a_i \leq 30$ 。

时间限制：2S

空间限制：256MB

D.终焉之排列

题面描述

CLB 有一个 $1 - n$ 的排列。

但是他想知道他手里的这个排列是不是一个“终焉之排列”。

怎么定义一个排列是否是“终焉”的方法如下：

如果在该排列中**存在**两个数 a, b , 且 $a + b$ 为偶数, 并且 $\frac{a+b}{2}$ 在排列中的位置在 a, b 之间, 则称这个排列为一个“终焉之排列”。否则若**不存在**这样的 a, b , 则不是一个“终焉之排列”。

于是 CLB 来询问你他手里的数列是否为一个“终焉之排列”。如果是输出 “YES”, 否则输出 “NO”。

输入格式

该题采用多组数据测试：

第一行包括一个正整数 T 表示数据组数：

之后包括 T 组数据，每一组的格式如下：

第一行一个正整数 n ，表示排列的长度。

第二行包括 n 个互不相同的且均在 $1 - n$ 范围的数用空格隔开，描述这个排列。

输出格式

对于每组数据输出一行一个字符串，“YES” 或者 “NO”（不含引号）。

样例

输入

```
2

5

1 5 2 4 3

4

1 3 4 2
```

输出

```
YES

NO
```

样例解释。

第一组：2在(1,3)之间, 4在(5,3)之间, 均满足。

第二组：没有这样的 a, b 。

数据规模

对于 10% 的数据: $1 \leq n \leq 100$ 。

对于 40% 的数据: $1 \leq n \leq 1000$ 。

对于 100% 的数据: $1 \leq n \leq 300000, 1 \leq T \leq 10$ 。

时间限制：2S

空间限制：256MB