

1 aria 的礼物

首先要判断字符串长度和相同部分的长度。

- $k \leq n/2$, 直接判断前 k 位与后 k 位有几位不同即可。
- $k > n/2$, 说明有重叠, 那最终修改过后的串应该是开头以 $n - k$ 为循环的循环串。那么对于一个位置来说, 每隔 $n - k$ 字母都应该是一样的, 则我们统计每个应该相同的位置的出现次数最多的字符, 然后把不同的都计入答案即可。

时间复杂度 $O(26 \times n)$ 。

2 神奇位运算

显然每一位都是独立的, 答案就是每一位各自需要的 01 串的数量的最大值。

我们分别考虑每一位:

如果要区分 and 和 or, 需要 0 和 1, $0 \text{ and } 1=0, 0 \text{ or } 1=1$;

区分 and 和 xor, 也需要 0 和 1, $0 \text{ and } 1=0, 0 \text{ xor } 1=1$;

区分 or 和 xor, 需要 1 和 1, $1 \text{ xor } 1=0, 1 \text{ or } 1=1$ 。

所以每一位至少需要 1 个 0, 2 个 1。如果原来的串中不满足该条件则需要在补充的 01 串上面补充对应缺少的 01。

所以答案就是 $\max_{i=1}^n \{\max(2 - \text{第 } i \text{ 位是 } 1 \text{ 的盘子数量 } 0) + \max(1 - \text{第 } i \text{ 位是 } 0 \text{ 的盘子数量 } 1)\}$ 。

复杂度 $O(n \times m)$ 。

3 终焉之数

该题暴力可以获得 50 分, 写的好的话。

状态压缩 dp。

设状态为 $f[i][sta]$ 表示在 b 中前 i 个数中, 出现过的因子状态为 sta , 的最小绝对值总和。

首先我们发现你最多会使用 16 个因子 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53。原因是如果你要用到 59, 不如把数变成 1, 代价永远不亏。所以你的 b 中数的范围为 $1 - 58$, 且你只会使用这 16 个因子。则 sta 为一个状态压缩态, 2^{16} 去表示每个因子在前 i 个数中是否出现过。

转移的时候枚举最后一位的数 $b[i]$, 假设 $b[i]$ 的分解因子状态为 $div[b[i]]$, 则我们可以枚举该位取什么数, 通过转移方程:

$$f[i][sta] = \min_{b[i]=1}^{58} \{f[i-1][sta \oplus div[b[i]]] + |a[i] - b[i]|\}.$$

PS: 某个你不认识的符号视作 xor。

去完成整个 dp 过程。至于 $div[b[i]]$ 可以预处理。整个复杂度为 $O(2^{16} \times n \times 60)$ 。

这样容易由于常数不够优秀，只能获得 70 分。

如果你这么实现没有获得 100 分，则你可以尝试如下优化：

- 小优化 1: 58, 57, 56, 54, 52, 51, 50, 48, 46, 45, 44, 42, 40 都用不到，可以减去一大部分常数。
- 小优化 2: 你可以预处理一个 $can[sta]$ 的数组，表示只用 sta 所表示的这些因子能形成 1-58 中的哪些数。这样在枚举 $b[i]$ 的时候可以排除掉很多冗余答案。

加上这两个优化必然获得满分。

4 终焉之排列

权值线段树 + 字符串 hash。

我们从左到右枚举中间数 $a[i]$ 。同时我们维护一个数组 H 。

$H[j] = 1$ 表示 j 在 $1 - i$ 中出现过。

$H[j] = 0$ 表示 j 在 $1 - i$ 中没出现过（说明应该在 $(i + 1) - n$ 中出现）。

那么对于中间数 $a[i]$ ，令 $len = \min\{a[i] - 1, n - a[i]\}$ 。

则我们要判断是否有以 $a[i]$ 为中间数的 (a, b) 对，则就是判断 $H[a[i] - len, a[i] - 1]$ 和 $H[a[i] + 1, a[i] + len]$ 的情况。

$H[a[i] - len, a[i] - 1]$ 和 $H[a[i] + 1, a[i] + len]$ 我们都可以看成 01 串。

如果 $H[a[i] - len, a[i] - 1]$ 和 $H[a[i] + 1, a[i] + len]$ 的倒序串完全相同，则说明不存在需要的 a, b 对。否则有不一样的说明，那个不一样的位置就是对应要的 a, b 对。

那么如何快速判断两个字符串相等，那只能使用字符串哈希（不会的同学可以自行百度博客上学习，本质上是取小素数 c ，和大素数 p ，将字符串携程 c 进制数然后对大素数 p 取模得到 hash 值。通常认为 hash 冲突概率比较小，则若 hash 值相同则串相同）。

同时我们要用线段树去维护 H 数组区间的字符串 hash 值。不过要维护两个，一个是正序 hash 值，一个是逆序 hash 值。然后从左往右枚举的同时修改线段树，并且查询对应线段的 hash 值是否相同。

时间复杂度 $O(n \log n)$