

CF Edu 189 B. Alternating String — 复盘对话录

zzy & Claude

2026-05-02

背景：题已 AC ([submission 373228609](#))，代码核心是数 相邻相等对子 (bad pair) 的个数 cnt ，判 $\text{cnt} \leq 2$ 即 YES。比赛时手滑写成 > 20 而非 > 2 WA 一发。这次复盘是要把“猜对了一半”的直觉补完，让 $\text{cnt} \leq 2$ 这条结论从结构上讲清楚——也就回答了它为什么只值 800。

第 1 步：题意 + 数据范围核对

Claude

先把题面 + 约束钉一下，确认我们在同一频道：

- 串 s 只含 a/b，长度 $n \in [2, 2 \cdot 10^5]$ ， $\sum n \leq 2 \cdot 10^5$ ，时限 2s。
- 至多一次操作：选区间 $[l, r]$ ，可选地把 $[l, r]$ 内每个字符 $a \leftrightarrow b$ (**invert**)，然后必须把 $[l, r]$ 整段反转 (**reverse**)。
- 问最后能否变成 alternating (任意相邻两位不同)。

数据规模直接给了线性时间预算，签到题量级。先确认这条阅读没问题，再走第 2 步——**为什么数 bad pair 就够了**。

第 2 步：抓不变量

Claude

先把第二步 invert 暂时放一边，只看 reverse 这一动作。

定义 **bad pair**：位置 i 满足 $s_i = s_{i+1}$ 。把 s 里所有 bad pair 按下标 i 分三类：

- 类 P： $l \leq i \leq r - 1$ (这对的两端都在被反转的段内)
- 类 Q： $i = l - 1$ 或 $i = r$ (恰好骑在被反转段的**边界**上)
- 类 R： $i < l - 1$ 或 $i > r$ (这对的两端都在被反转段外)

问题：reverse 之后，这三类位置里哪一类的“等 / 不等”状态**铁定不变**？哪一类**有可能变**？(拿 abbaba 这种串选一段在纸上反一下，盯紧每个相邻对反转前后是否还相等，比脑算靠谱。)

zzy

如果两段都在反转段内，那么是不变的。如果两段刚好骑在反转边界上，那就是比较 trick，不知道是什么情况，你也不知道反转弯会怎么样，因为这和段有关系。如果两段都在反转外，肯定不变。当然，如果在反转段的边界上有办法解决，因为它还有操作可以反转。如果反转段反转后还一样，就再反转一下字符就行了。说白了，只要有一个在段内，那么这个 bad pair 肯定是可以被消掉的不对不对，只有在这个**边界**上，它才可以被消掉的。

Claude

对。三类的状态分别是：

- **P 类 (内部)**：reverse 把 (s_i, s_{i+1}) 翻成 (s_{i+1}, s_i) ，相等性守恒；invert 同时翻 a/b，相等性也守恒。一定保留。
- **R 类 (外部)**：完全没碰。一定保留。
- **Q 类 (边界)**： s_{l-1} 旁边的字符从 s_l 换成（也许 invert 过的） s_r ； s_{r+1} 旁边的字符从 s_r 换成（也许 invert 过的） s_l 。有可能变。

立刻得到**必要条件**：原串里所有 bad pair 必须挤在 $\{l-1, r\}$ 这两个位置上才有救——也就是 $\text{cnt} \leq 2$ 。

这就是你 AC 那条 if 的”必要”那一半，也是这道题为什么定到 800：核心就一个不变量。剩下”充分”那一半，下一步走。

第 3 步：充分性收尾

zzy

讲到这里题就做完了，充分性这一半不用再细抠，结论我自己脑算就行。

Claude

那 mentor 我直接把充分性收掉，留个完整记录：

设两个 bad pair 在位置 $p_1 < p_2$ ，要让 $\{l-1, r\} = \{p_1, p_2\}$ ，所以 $l = p_1 + 1$ ， $r = p_2$ ，被操作的子串是 $s[p_1 + 1..p_2]$ ，硬选定。

操作后 boundary 字符：

- 位置 p_1 旁的新字符 = s_{p_2} 或 $\neg s_{p_2}$ （取决于是否 invert）；
- 位置 p_2 旁的新字符 = $s_{p_1+1} = s_{p_1}$ 或 $\neg s_{p_1}$ （同上）。

需要 $s_{p_1} \neq$ 新字符且新字符 $\neq s_{p_2+1} = s_{p_2}$ ，化简后两个边界条件殊途同归，都是同一句” s_{p_1} 和 s_{p_2} 关系”：

- $s_{p_1} \neq s_{p_2}$ （一个 aa、一个 bb，**异色**）：reverse 单刷就够，boundary 字符互换正好掐准；
- $s_{p_1} = s_{p_2}$ （两个都 aa 或都 bb，**同色**）：reverse + invert，invert 把字母翻一下 boundary 才不等。

两种情形都覆盖 $\Rightarrow \text{cnt} = 2$ 一定 YES。cnt = 1 把上述退化（让一边 boundary 落在串端外，只剩一个边界要修，同样一定能成）。cnt = 0 trivial。

综合： $\text{cnt} \leq 2 \iff \text{YES}$ ，这就是题解里那条 if 的全部底气。这道题为啥定 800：核心只有一个不变量——**reverse + invert 在「相邻相等」这件事上对内部和外部都是恒等，只有两个边界能动**。剩下的全是把这个不变量翻译成代码。

复盘要点：

- **比赛 bug**：把 $\text{cnt} > 2$ 写成 $\text{cnt} > 20$ ——签到题这种字面 typo 得养成 submit 前肉眼扫一遍 cutoff 数字的习惯；
- **思考过程**：你”猜到了一半”对应的就是必要侧（边界能动），充分侧的 case-work 没顶住单纯是因为已经验过样例所以没回头补；
- **知识点**：操作不变量类签到题——遇到「最多一次 reverse / swap / 区间操作」立刻去找”什么量在操作下守恒”，cnt 这种”边界数”是高频答案。