

产品检测 命题报告

using

2025 年 9 月 26 日

目录

1	题目大意	2
1.1	题目描述	2
1.2	评分规则 and 数据范围	2
2	解题过程	4
2.1	解法 0	4
2.2	解法 1	4
2.3	解法 2	4
2.3.1	解法 2.1	4
2.4	解法 3	4
2.5	解法 4	5
2.5.1	解法 4.1	5
2.5.2	解法 4.2	5
2.6	解法 5	5
2.6.1	解法 5.1	5
2.7	解法 6	5
3	参考资料	7

1 题目大意

1.1 题目描述

世界上最伟大的公司，lezeron 公司，找你定制一个奇怪的机器。

lezeron 公司搭建了一条流水线，生产一种价值 50 Coins 的产品。然而，由于流水线刚刚建成，难免会产生一些次品，你的机器需要找出这些次品。

流水线在搭建的时候考虑到了次品的问题，所以每个产品生产时都有一个 01 串编码，检查产品只需要检查：

- 初始产品编码的 0 和 1 数量相同。
- 初始产品编码的每一个前缀 0 的数量均不少于 1 的数量。

但同样由于产品的形态，性质等问题，你的机器在检查产品时有很大的限制。具体的说，你的机器由 n 个检测仓构成，每个检测仓都是下面两者之一：

- 读取仓：读取并删除产品编码的第一个字符（如果存在），根据读取到的字符（或者没有），将产品运送到另一个检测仓。
- 追加仓：在产品编码末尾追加任意一个字符，并将产品运送到某个检测仓。

你的检测仓由 $1 \sim n$ 编号，新生产的产品会出现在检测仓 1。还有两个编号特殊的检测仓，0 和 -1 ，它们不属于上面两种，具体的，运送到 0 的产品会被标记为“合格”，运送到 -1 的产品会被标记为“次品”。

由于资金有限，你的检测仓不能超过 302 个，即 $1 \leq n \leq 300$ 。除了字符 0 和字符 1 外，你的追加仓还可以写其它字符，具体的，你可以设定一个字符集 m ，你的追加仓可以写 $0 \sim m - 1$ 内每一个字符。由于可用字符有限，你需要保证 $2 \leq m \leq 128$ 。

由于你的机器是需要有作用的，你的机器不能让同一个产品被运送超过 10^6 次。如果产品在机器里待了过长时间，lezeron 公司的维修人员就会关闭机器取出产品，并将你的这次检测结果标记为“错误”。

当然，由于 lezeron 公司口碑极佳，少量的检测错误并不会带来很大的影响，因此，lezeron 公司允许你的机器有一定的出错率。当然完美的机器总是更好的，你的机器越优秀，你得到的薪酬越高。

lezeron 公司计划了 10 种场景，每种场景完全独立。lezeron 公司对你的机器的评价就是每个场景你的机器的表现的和。设计更优秀的机器，拿到更高的薪酬（分数）！

1.2 评分规则 and 数据范围

本题一共 10 个测试点，每个测试点 10 分，对应 10 个场景，每个场景都有 100 个产品用于测试你的机器。集合 $S = \{67, 75, 84, 88, 92, 93, 95, 98, 99, 100\}$ ，对于每个测试点，若你的机器正确检测了 k 个产品，对于任意 $x \in S$ ，若 $k \geq x$ ，在这个测试点上你都可以获得一分。

样例输出在每个场景上都可以正确检测 61 个产品，因此本样例可以获得 0 分。
对于每个测试点，令所有初始产品编码的长度最大为 L ，有：

测试点编号	$L \leq$	测试点编号	$L \leq$
1	2×10^2	6	3×10^4
2	1×10^3	7	5×10^4
3	1.2×10^3	8	3×10^5
4	5×10^3	9	9×10^5
5	1×10^4	10	9.5×10^5

2 解题过程

造一个自动机，判断字符串是否为匹配括号串。以下约定 $N = 300$ 为检测仓最大数量限制。

2.1 解法 0

将所有产品标记为“合格”。

期望得分：1。

2.2 解法 1

猜测出题人不会造数据，不合格的数据都是随的，所以计算 0 和 1 的数量差模 300 的结果，如果为 0 就标记为“合格”。

有点小聪明，但不多。

期望得分：23。

2.3 解法 2

尝试用所有的节点模拟栈，可以解决 $L \leq 200$ 的部分。

爆栈之后可以假定字符串不合格。

期望得分：25。

2.3.1 解法 2.1

尝试融合解法 1 和解法 2，选择 K ，当栈深度低于 K 时采用解法 2，大于等于 K 之后换成解法 1。

栈深 K ，模数 $N - K$ ， $K = 200$ 可以获得不错的分数。

期望得分：31。

2.4 解法 3

用栈模拟这个太粗浅了，考虑一些括号序列别的判断方法：每次删除一对相邻匹配的括号，如果可以删空，那么它就是一个匹配的括号串。

向字符串末尾加一个分隔符，设置一个大小为 2 的双端队列，任意时刻队列空间为 01，清空队列，否则如果队列空间不够，将队列头字符放到分隔符后腾位置。

显然，代价最大的字符串是 000...111... 的样式，每次循环遍历一轮字符串，插入一轮字符串，遍历后字符串长度减 2，时长约为 $\frac{L^2}{2}$ ，可以解决 $L \leq 1.2 \times 10^3$ 的部分。

期望得分：31。

2.5 解法 4

队列大小只有 2 简直太浪费了，我们还有那么多检测仓能用。

使用类似解法 2 的方式扩大队列，这样一次就可以检测 $O(N)$ 层了。

时长约为 $\frac{L^2}{2N}$ ，可以解决 $L \leq 10^4$ 的部分。

期望得分：52。

2.5.1 解法 4.1

解法 4 做法的在其它测试点无法拿到很多分的根本原因是超时。

尝试通过限制运行轮数减少超时可能，可以通过改变分隔符记录运行轮数，超过运行轮数后标记为“不合格”。

$m = 22$ 可以获得不错的分数，尽管在前 5 个测试点中会损失一些，但后面测试点的收获足以弥补。

期望得分：55。

2.5.2 解法 4.2

3×10^4 相比 10^4 仅仅 3 倍，如果能压缩一下字符串的长度也许会舒服一些。

双端队列左侧需要弹出 0 的时候，我们可以尝试一次弹出 K 个 0，打包成一个新字符，这样第一轮后所有的 0 全部被打包，大大减少了读取的字符串长度。

这个新字符没有使原来的方案变复杂很多。 $K = 7$ 可以获得不错的分数。

期望得分：59。

2.6 解法 5

解法 4.2 基础上，如果我们还能打包字符 1，那就再好不过了。

实际上这是可行的，我们可以再开一个双端队列存储字符 1 的计数。根据其存储的字符，我们将两个双端队列分别称为 q_0 和 q_1 。

需要注意的是压缩 0 的插入在 q_0 填满清空之时，而压缩 1 的插入在 q_1 由空到非空之时。

被压缩的 01 仍然保留了原来的 01 的效用，因此可以进行更多的压缩。令队列大小限制为 K ，每轮之后字符串长度由 L' 限制变为 $\frac{L'-c}{K}$ ，其中 c 为本轮因匹配而被删除的字符数。

结点数是 $K^2 + K + 3$ ，所以 K 最大为 16，忽略 c ，时长约为 $\frac{17L}{15}$ 。期望得分：81。

2.6.1 解法 5.1

同解法 4.1，解法 5 同样有超时的问题，那么我们也可以试图通过限制轮数解决。

然而遗憾的是，由于解法 5 的字符串长度收敛过快，对于后两个测试点，它们会在很早的轮数超时，而对于前面的测试点，限制轮数对它们的损失过大，所以这个策略并不会带来额外的分数。

$m = 10$ ，尽管这和不限制轮数没有任何区别。期望得分：81。

2.7 解法 6

K^2 的节点数太大了，考虑降低这个值，即，考虑只用一个量同时维护两个值。

如果两个值没有联系，这是不可能的，但幸运的是，这两个值并非毫无联系。我们注意到， q_1 增加元素需要 q_0 为空，这启发我们只记录两个队列中元素数量的差 $d = |q_0| - |q_1|$ 。

解法 4.2 告诉我们，对于 q_0 ，我们不强制在队列内元素恰好 K 个时压缩，可以在多余 K 个时只压缩其中 K 个，也就是说，我们可以允许在 $d \geq K$ 时压缩 0。

然而，对于一个压缩后的 10 结构，它的原来的组成方式有可能是 110001、100110 等等类似的乱七八糟的形式，不过没关系，括号序列有一种定义是“0 和 1 数量相同且每一个前缀 0 的数量均不少于 1 的数量”，这个 10 结构展开后不会有 0 在前 1 在后的情况（因为这对 01 会在压缩前抵消），证明它们的充要关系是简单的。

结点数降至 $3K + 4$ ，所以 K 最大为 98，忽略 c ，时长约为 $\frac{99L}{97}$ 。期望得分：100。

3 参考资料

- QOJ4553: <https://qoj.ac/problem/4553>, 本题采用的是 Task4 的部分, 并细化了评分方式。