QTA 第三周笔试题解答

欧岱松

2025 年 12 月 20 日

1. 骰子问题

题目：使用一个标准的 $1 \sim 6$ 骰子，构造出 $1 \sim 7$ 的均匀分布。
解答：使用拒绝抽样法（Rejection Sampling）。

1. 投掷骰子两次，得到结果 a 和 b（$1 \leq a, b \leq 6$）。
2. 计算索引值 $val = (a-1) \times 6 + b$。这将生成 $[1, 36]$ 区间内的均匀整数。
3. 如果 $val \leq 35$：结果为 $(val - 1) \bmod 7 + 1$。
4. 如果 $val = 36$：舍弃该结果，从第 1 步重新开始。

由于 35 是 7 的倍数，每个数字 1 到 7 在接受条件下出现的概率均为 $5/35 = 1/7$，因此该方法生成的分布是均匀的。

2. 截面问题

题目：圆锥的截面是什么形状？
解答：假设切面平行于底面，截面形状为**圆形**。

- **证明**：设圆锥沿 z 轴对齐，其方程为 $x^2 + y^2 = c^2 z^2$。平行于底面的截面意味着 $z = k$（常数）。将 k 代入方程，得到 $x^2 + y^2 = (ck)^2 = R^2$，这描述的是一个圆。
- **注**：如果平面倾斜，截面将产生圆锥曲线，可能是椭圆、抛物线或双曲线，证明暂时没有写出来。

3. 数字序列问题

题目：能否将数字 $1, 1, \ldots, 2026, 2026$ 排列，使得两个 k 之间恰好有 k 个数字？
解答：**不能。**

这是一个 Langford 配对问题。对于数字 n，存在这样排列的必要条件是：

$$n \equiv 0 \pmod 4 \quad \text{或} \quad n \equiv 3 \pmod 4$$

检验 $n = 2026$：

$$2026 = 4 \times 506 + 2 \implies 2026 \equiv 2 \pmod 4$$

由于不满足必要条件，因此这样的排列不存在。

必要条件的证明：

假设 Langford 序列长度为 $2n$，所有位置下标 $\{1, 2, \ldots, 2n\}$ 的总和为：

$$S = \sum_{i=1}^{2n} i = n(2n+1) = 2n^2 + n$$

对于数字 k，设其两个位置分别为 x_k 和 y_k，由题意有 $y_k = x_k + k + 1$（两个 k 之间恰好有 k 个数字）。

因此，数字 k 的两个位置之和为：

$$x_k + y_k = x_k + (x_k + k + 1) = 2x_k + k + 1$$

对所有 $k \in \{1, 2, \ldots, n\}$ 求和，得到位置总和的另一种表达：

$$S = \sum_{k=1}^{n}(2x_k + k + 1) = 2\sum_{k=1}^{n} x_k + \sum_{k=1}^{n} k + \sum_{k=1}^{n} 1 = 2\sum_{k=1}^{n} x_k + \frac{n(n+1)}{2} + n$$

联立两个 S 的表达式：

$$2n^2 + n = 2\sum_{k=1}^{n} x_k + \frac{n(n+1)}{2} + n$$

化简得：

$$2n^2 = 2\sum_{k=1}^{n} x_k + \frac{n(n+1)}{2}$$

移项：

$$2\left(n^2 - \sum_{k=1}^{n} x_k\right) = \frac{n(n+1)}{2}$$

由于左边显然为偶数，因此右边的 $\frac{n(n+1)}{2}$ 也必须是偶数，即 $n(n+1)$ 必须能被 4 整除。

由于 n 和 $n+1$ 是连续整数，其中必有一个是偶数。要使 $n(n+1) \equiv 0 \pmod 4$，需要：

- 若 n 是偶数，则 $n \equiv 0 \pmod 4$
- 若 n 是奇数，则 $n+1$ 必须被 4 整除，即 $n \equiv 3 \pmod 4$

因此，Langford 序列存在的必要条件为：

$$n \equiv 0 \pmod 4 \quad 或 \quad n \equiv 3 \pmod 4$$

4. 蓝眼睛问题

题目：一个岛上有 100 个蓝眼睛的人和 900 个棕眼睛的人。岛上的规则是：如果有人知道自己是蓝眼睛，就必须在当晚自杀。某天，一个外来者说：”我看到了一个蓝眼睛的人。”会发生什么？

解答： 第 100 天晚上，所有 100 个蓝眼睛的人都会自杀。

数学归纳法：

- **基础情况：**如果只有 1 个蓝眼睛的人，他听到外来者的话后，看到其他人都是棕眼睛，就知道自己是蓝眼睛，第 1 天晚上自杀。
- **归纳步骤：**假设有 N 个蓝眼睛的人。每个蓝眼睛的人看到其他 $N-1$ 个蓝眼睛的人。如果前 $N-1$ 天没有人自杀，他们就能推断出蓝眼睛的人数不是 $N-1$，而是 N（包括自己），因此第 N 天晚上所有蓝眼睛的人都会自杀。

5. 猜牌问题

题目：Bob 从一副 52 张牌中抽取 5 张，向 Alice 展示其中 4 张。Alice 能否根据这 4 张牌推断出第 5 张牌？

解答：**可以。**

策略：

1. **确定花色（鸽笼原理）**：5 张牌中至少有 2 张花色相同。Bob 选择其中一张作为隐藏牌，另一张作为第一张展示的牌，以此传递花色信息。
2. **编码点数**：剩余 3 张牌可以按 $3! = 6$ 种不同方式排列。Bob 计算隐藏牌与展示牌之间的距离 d（模 13），选择使 $d \leq 6$ 的方向。然后用 3 张牌的排列顺序编码这个距离 d。Alice 根据后 3 张牌的排列顺序解码出距离 d，将第一张牌的点数加上 d（模 13）即可得到隐藏牌。

6. 盛最多水的容器

题目：给定一个整数数组 $height$，其中 $height[i]$ 表示第 i 条垂直线的高度。找出两条线，使得它们与 x 轴构成的容器能盛最多的水。

解答：使用**双指针算法**，时间复杂度为 $O(N)$。

算法思路：

1. 初始化左右指针分别指向数组的起始和末尾。
2. 计算当前容器的面积 $= \min(height[left], height[right]) \times (right - left)$。
3. 更新最大面积。
4. 移动较短的那一边的指针（贪心策略）。
5. 重复步骤 2-4，直到两指针相遇。

代码实现：

```python
def maxArea(height):
    """
    求解盛最多水的容器问题
    参数: height - 表示每条垂直线高度的整数数组
    返回: 容器能够容纳的最大水量
    """
    # 初始化左右指针
    left, right = 0, len(height) - 1
    max_area = 0

    # 当两指针未相遇时循环
    while left < right:
        # 计算当前容器面积
        current_height = min(height[left], height[right])
        current_width = right - left
        current_area = current_height * current_width

        # 更新最大面积
        max_area = max(max_area, current_area)

        # 贪心策略: 移动较短的一边
```

```python
        if height[left] < height[right]:
            left += 1
        else:
            right -= 1

    return max_area


# 测试示例
height = [1, 8, 6, 2, 5, 4, 8, 3, 7]
result = maxArea(height)
print("输入:", height)
print("输出:", result)
# 解释: 选择 height[1]=8 和 height[8]=7
#       面积 = min(8,7) * (8-1) = 7 * 7 = 49
```

Listing 1: 双指针算法 Python 实现