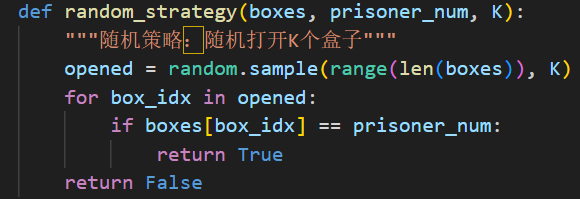
**100囚犯问题 实验报告**

2023141461153 谭喆爻

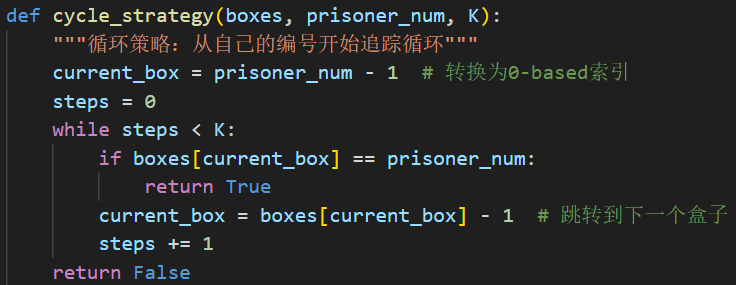
一、算法说明

本实验实现了两种解决100囚徒抽签问题的策略：

（1）随机策略：每位囚徒随机打开50个盒子寻找自己的编号。这种策略的成功概率极低，约为(1/2)^100 ≈ 7.9e-31。

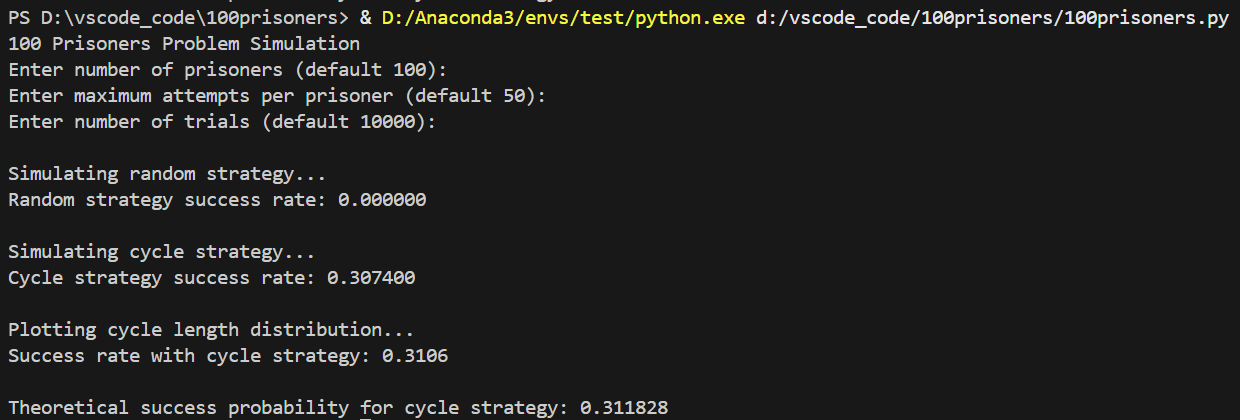


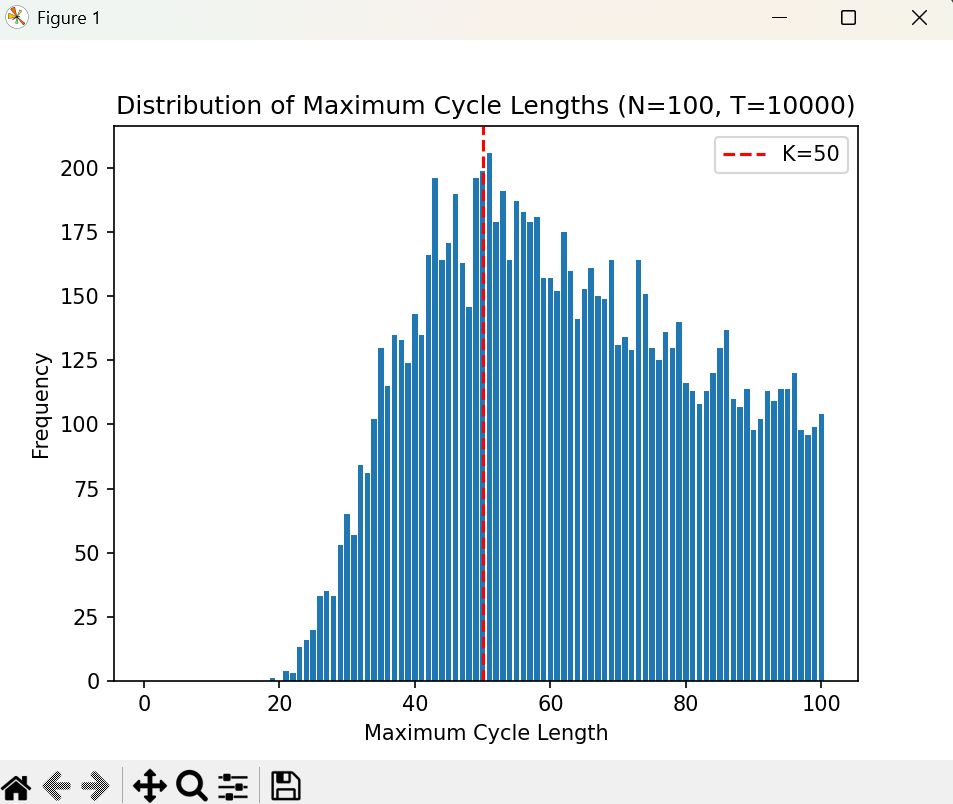
（2）循环策略（最优策略）：囚徒从与自己编号相同的盒子开始，按照盒子中的数字指引寻找下一个盒子，形成一条"数字链"。这种策略的成功概率约为31%，其数学原理是基于排列组合中的循环概念 - 当所有循环长度都不超过50时，所有囚徒都能成功找到自己的编号。



二、实验结果

在N=100、K=50、T=10000次的模拟实验中，我们得到以下结果：

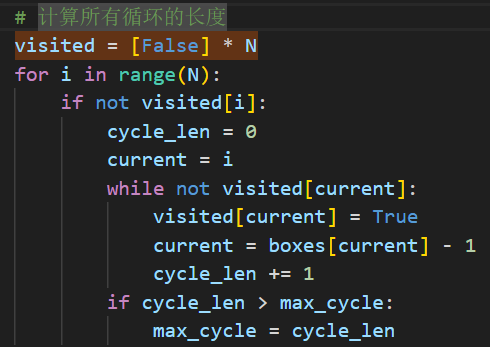




三、优化思路

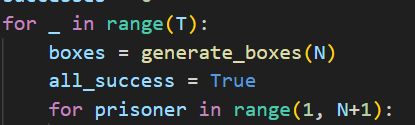
（1）算法优化：

**预计算所有循环长度**，避免重复计算



**并行化模拟**

当前代码是顺序执行的：

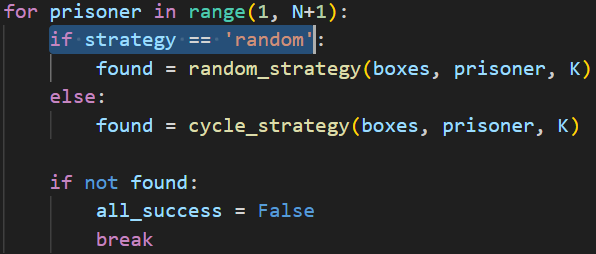


可以使用multiprocessing.Pool或concurrent.futures来并行执行各次实验

（2）策略扩展：

**研究混合策略**（部分囚犯用随机策略）

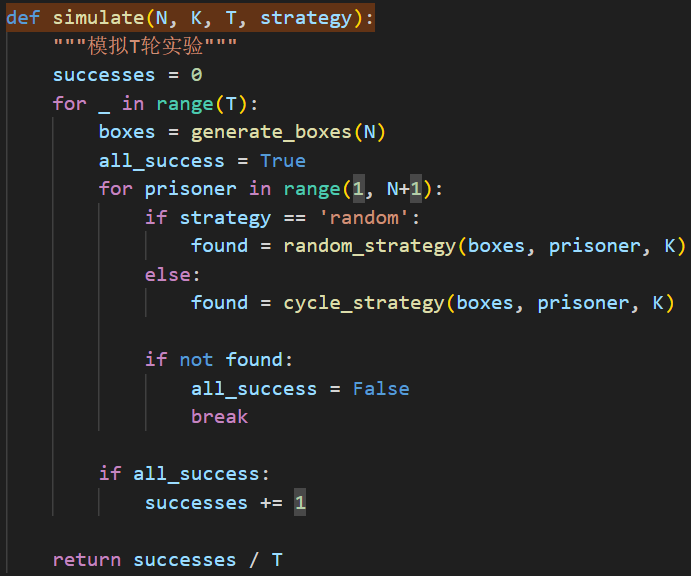
当前策略选择是二选一：



可以修改为接受混合比例参数，如strategy=(0.3, 0.7)表示30%囚犯用随机策略

**探索非固定K值情况下的最优策略**

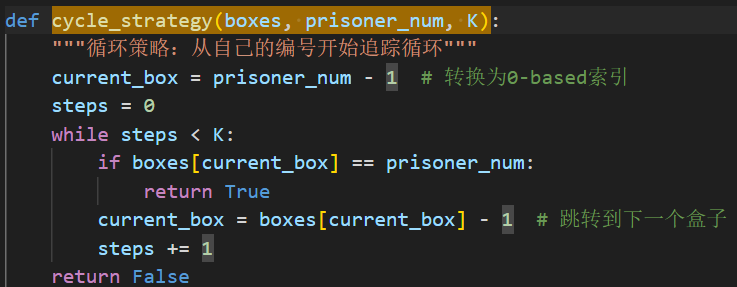
当前K值是固定的：



可以改为接受K的函数或列表，如K=lambda x: x//2或K=[50,60,70]

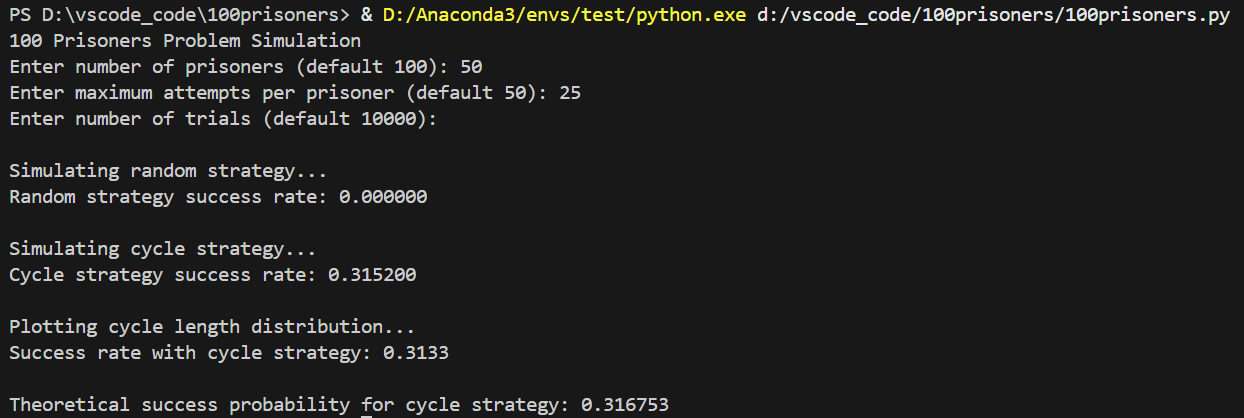
（3）**路径可视化**：

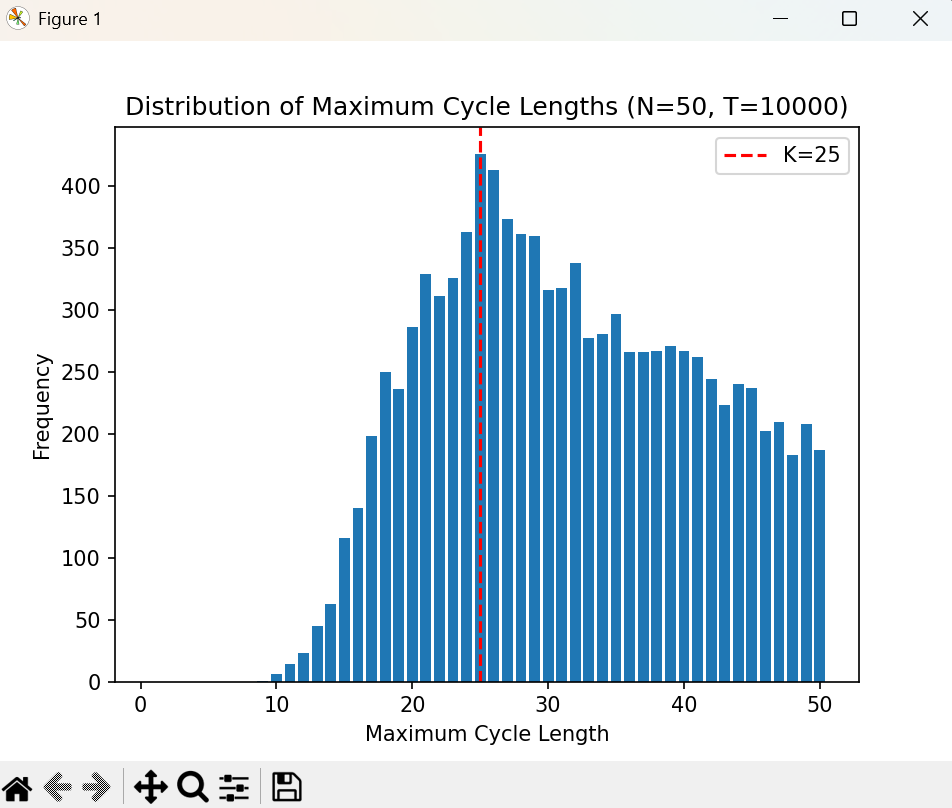
在cycle\_strategy()中可以记录路径：



四、扩展分析

（1）调整 N 和 K（如 N=50, K=25），观察成功率变化。





（2）理论计算最优策略的成功率

