**每日日志\_3（2025年7月2日）**

**一、今日已完成**

今日完成了里程碑的步骤三和步骤四。

步骤三：我认真阅读了本步骤的PDF电子书，了解了公钥与加密体系等知识点。

步骤四：这段代码的作用是从指定的 URL 获取一组数学计算题目，并通过快速幂算法（fastModular）逐一计算它们的结果。首先，程序向服务器发送 GET 请求获取包含计算题目的 JSON 数据，然后遍历这些题目，使用 fastModular 函数进行计算，最终将所有计算结果通过 GET 请求以参数的形式发送回服务器，并打印出程序的执行时间和服务器返回的结果。此代码实现了快速计算大数的幂模操作，并测量并优化了计算的时间效率。

**二、遇到的问题、心得体会**

在编写这段代码时，我遇到了一些问题并获得了新的体会。首先，由于 Python 中字符串是不可变的，频繁的字符串拼接会导致性能问题，尤其在处理大量数据时。以后我会考虑使用 list 来代替字符串拼接，拼接完成后再转回字符串，这样可以提高效率。其次，坐标转换在实现棋盘时容易出错，我通过 getIndex 函数将棋盘坐标转换为线性索引，但在实际应用中要特别注意坐标的有效性和边界问题。最后，我认识到在棋盘表示时，虽然用字符串表示棋盘简洁，但随着需求复杂化，应该逐步采用更高效的数据结构如二维数组，以提高代码的可扩展性和维护性。通过这段代码，我更深刻理解了性能优化、数据结构选择和 API 交互的实际应用。