**深 圳 大 学 实 验 报 告**

**课程名称： 互联网编程**

**实验项目名称： 多线程/线程池TCP服务器端程序设计**

**学院： 计算机与软件学院**

**专业： 计算机科学与技术**

**指导教师： 刘志丹**

**报告人： 郑雨婷 学号： 2021150122 班级： 高性能**

**实验时间： 2023/3/30 2023/5/5**

**实验报告提交时间： 2023/5/5**

**教务处制**

一、实验目的与内容：

**目的：**熟悉java线程编程技术，掌握线程技术在JAVA互联网通信程序中的应用。

**内容要求：**

1. 多线程TCP服务器（30分）：

设计编写一个TCP服务器端程序，需使用多线程处理客户端的连接请求。客户端与服务器端之间的通信内容，以及服务器端的处理功能等可自由设计拓展，无特别限制和要求。

1. 线程池TCP服务器（30分）：

设计编写一个TCP服务器端程序，需使用线程池处理客户端的连接请求。客户端与服务器端之间的通信内容，以及服务器端的处理功能等可自由设计拓展，无特别限制和要求，但应与第1项要求中的服务器功能一致，便于对比分析。

1. 比较分析不同编程技术对服务器性能的影响（10分）：

自由编写客户端程序和设计测试方式，对1和2中的服务器端程序进行测试，分析比较两个服务器的并发处理能力。

1. 设计编写可重用的服务器日志程序模块，日志记录的内容和日志存储方式可自定（比如可以记录客户端的连接时间、客户端IP等，日志存储为.TXT或.log文件等），分别在1和2的服务器程序中调用该日志程序模块，使多线程TCP服务器和线程池TCP服务器都具备日志功能，注意线程之间的同步操作处理。（20分）

注意：

1. 实验报告中需要有实验结果的截屏图像。

2. 需要现场验收，不得抄袭。

二、实验过程和代码与结果

**1. 给出满足内容要求1的程序源码及运行结果，简述思路或实验过程。**

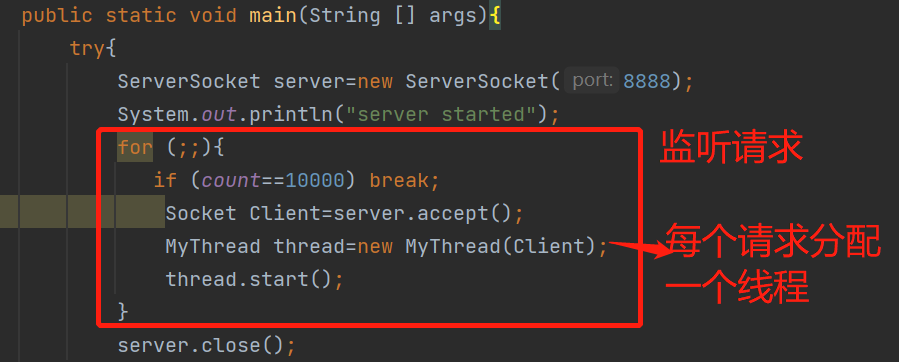
编写时间请求程序，客户端向服务器发送Time,服务端接收到请求显示系统时间。

思路：每增加一个连接，服务器就为该连接新建一个线程。因此，在服务端需要创建Thread的派生类MyThread，服务端中MyThread类代码如下：



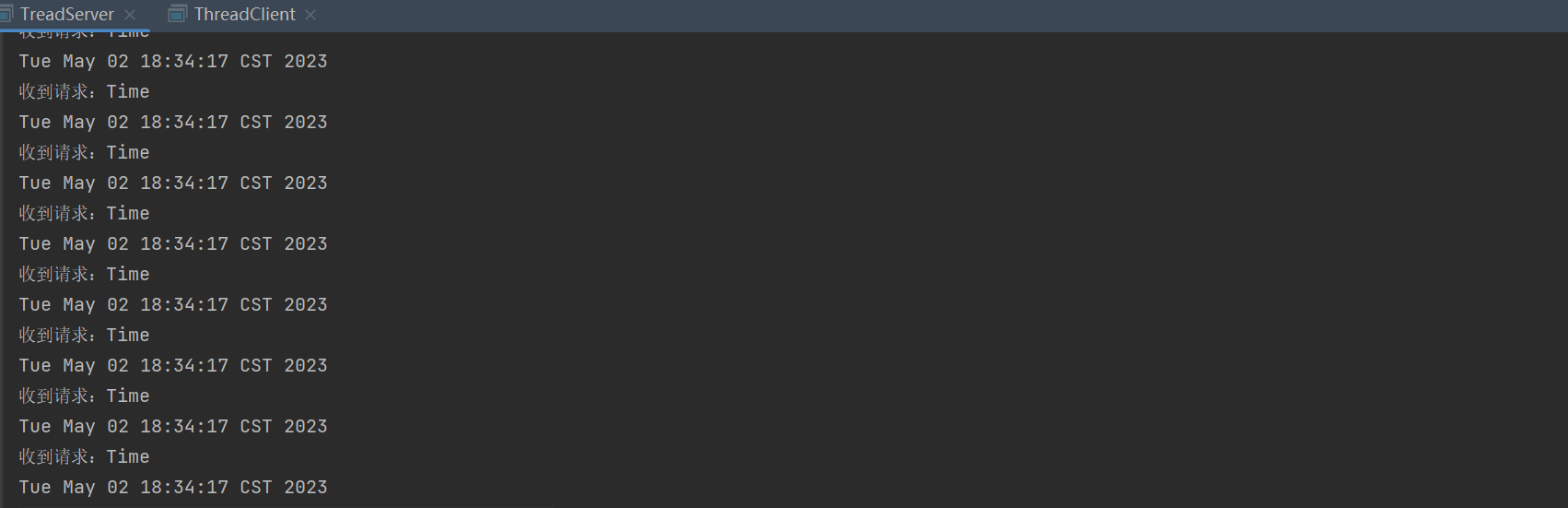
在每个线程的run()方法中：用socket.getInputStream()方法获取输入流，显示收到的请求。用Date来得到当前的系统时间nowTime并显示。

每收到一个请求，就建立一个新的线程，并start()开启线程，多线程服务器的主程序代码如下：



首先创建服务端套接字，在端口8888进行监听。

对服务器发起1000个连接，测试出结果如下，多线程响应连接正常。

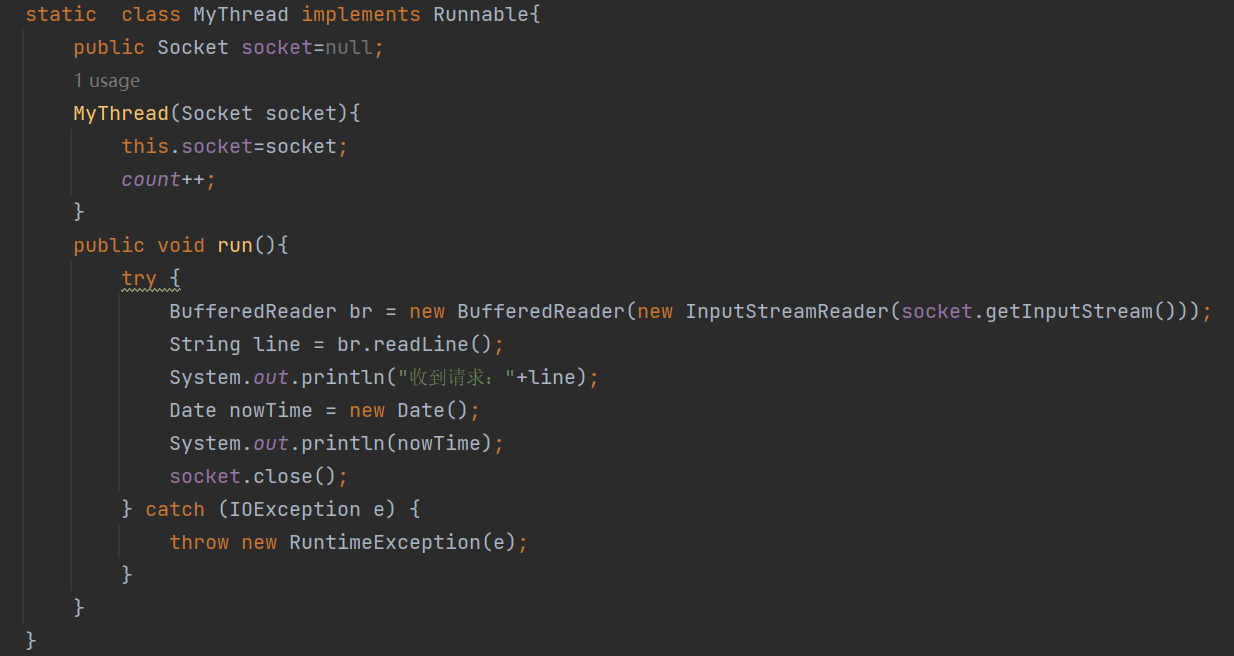


**2. 给出满足内容要求2的程序源码及运行结果，简述思路或实验过程。**

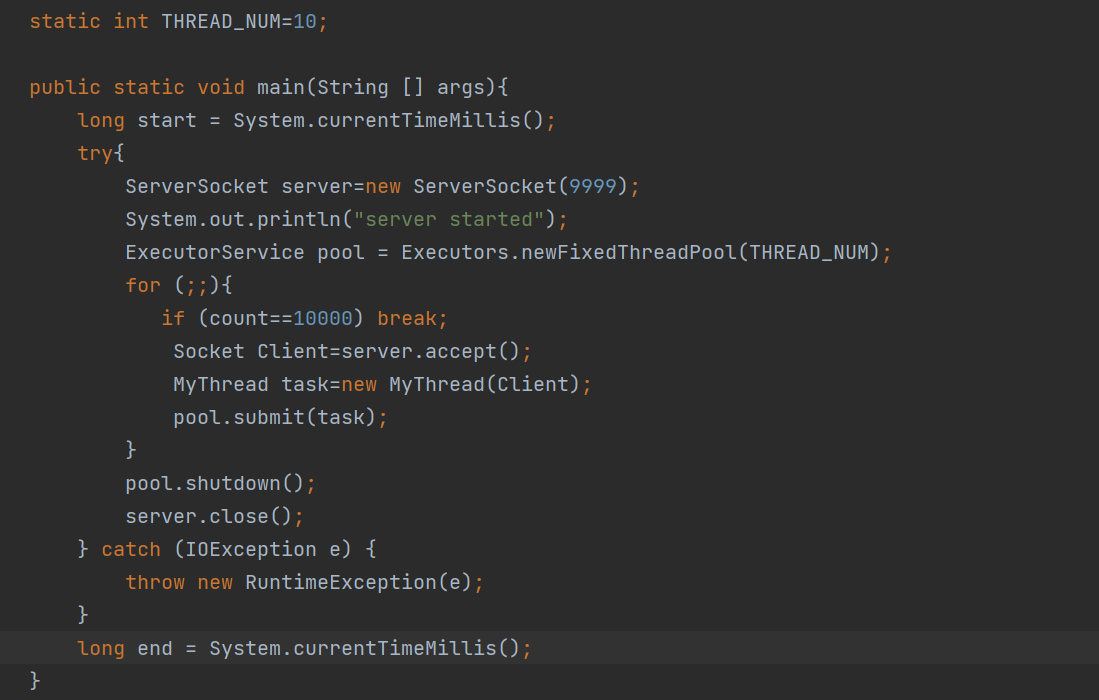
编写时间请求程序，客户端向服务器发送Time,服务端接收到请求显示系统时间。

思路：使用线程池，在服务器端使用线程池。创建实现Runnable接口的对象交给ExecutorService 执行,设置线程池数量为10。其余与1中相同。

将MyThread类由Thread的派生类改成实现Runnable接口的类：

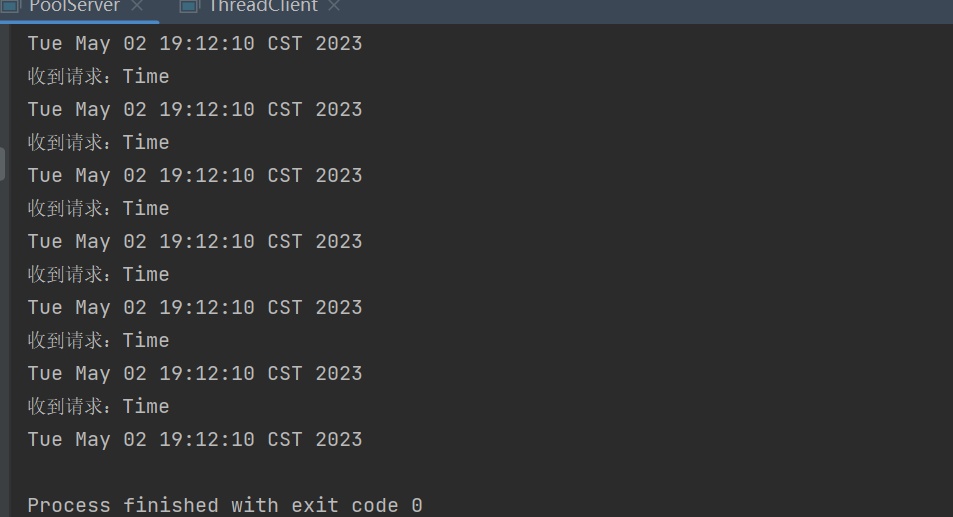


线程池服务端代码如下：



首先创建服务器套接字，，在端口9999进行监听。然后建立一个大小为THREAD\_NUM的线程池（THREAD\_NUM=10）每收到一个请求，创建一个对象直接交给 ExecutorService 执行。最后用shutdown()关闭线程池。

验证：对服务器发起1000个连接，测试出结果如下，线程池响应连接正常。



**3. 按内容要求3给出测试结果及分析对比情况，阐述测试方法或过程。**

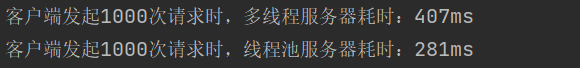
编写一个测试程序，首先编写一个task类，根据task类中的成员变量port来确定像哪个服务器发起连接，8888为多线程服务器，9999为线程池服务器。



分别对多线程服务器和线程池服务器发起多个连接请求，统计所需要的时间。只需改变NUM即可改变连接规模，代码如下：



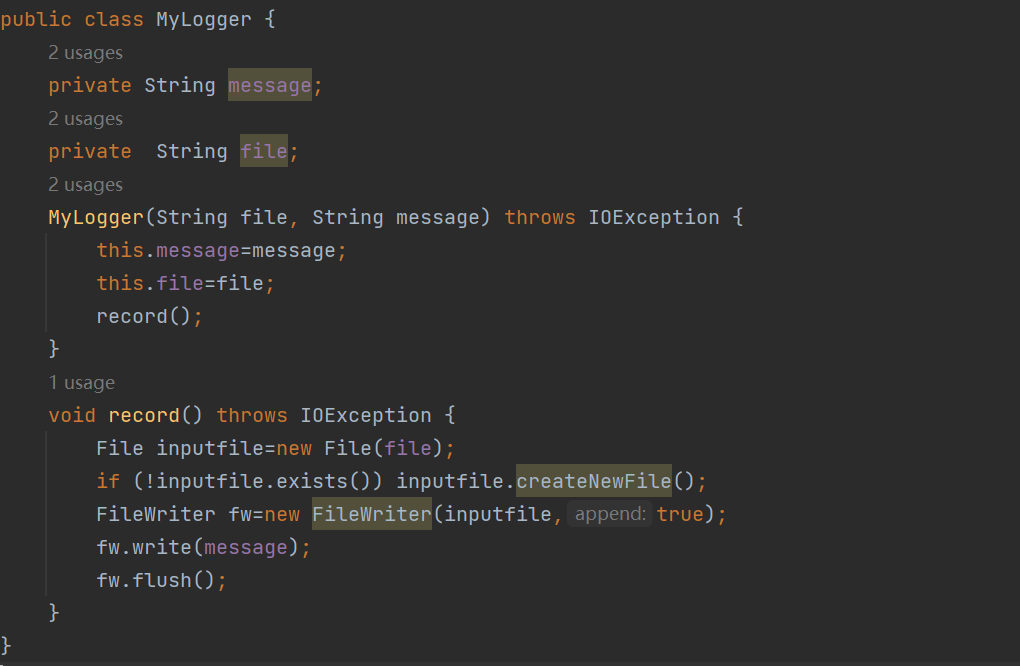
将NUM设置为1000进行测试，测试结果与预想的相同，验证该测试程序可行。



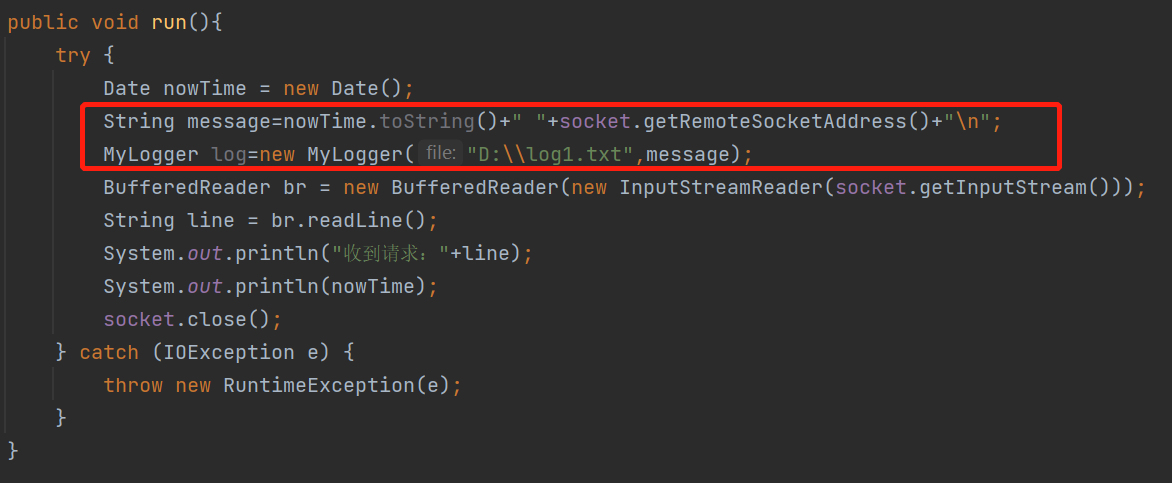
改变NUM为500-3500，得到的测试结果与1000次一致，都为多线程服务器耗时较长，线程池服务器耗时较短。

1. **给出满足内容要求4的实验结果，包括源码及两个服务器增加了日志功能后，日志记录运行结果。**

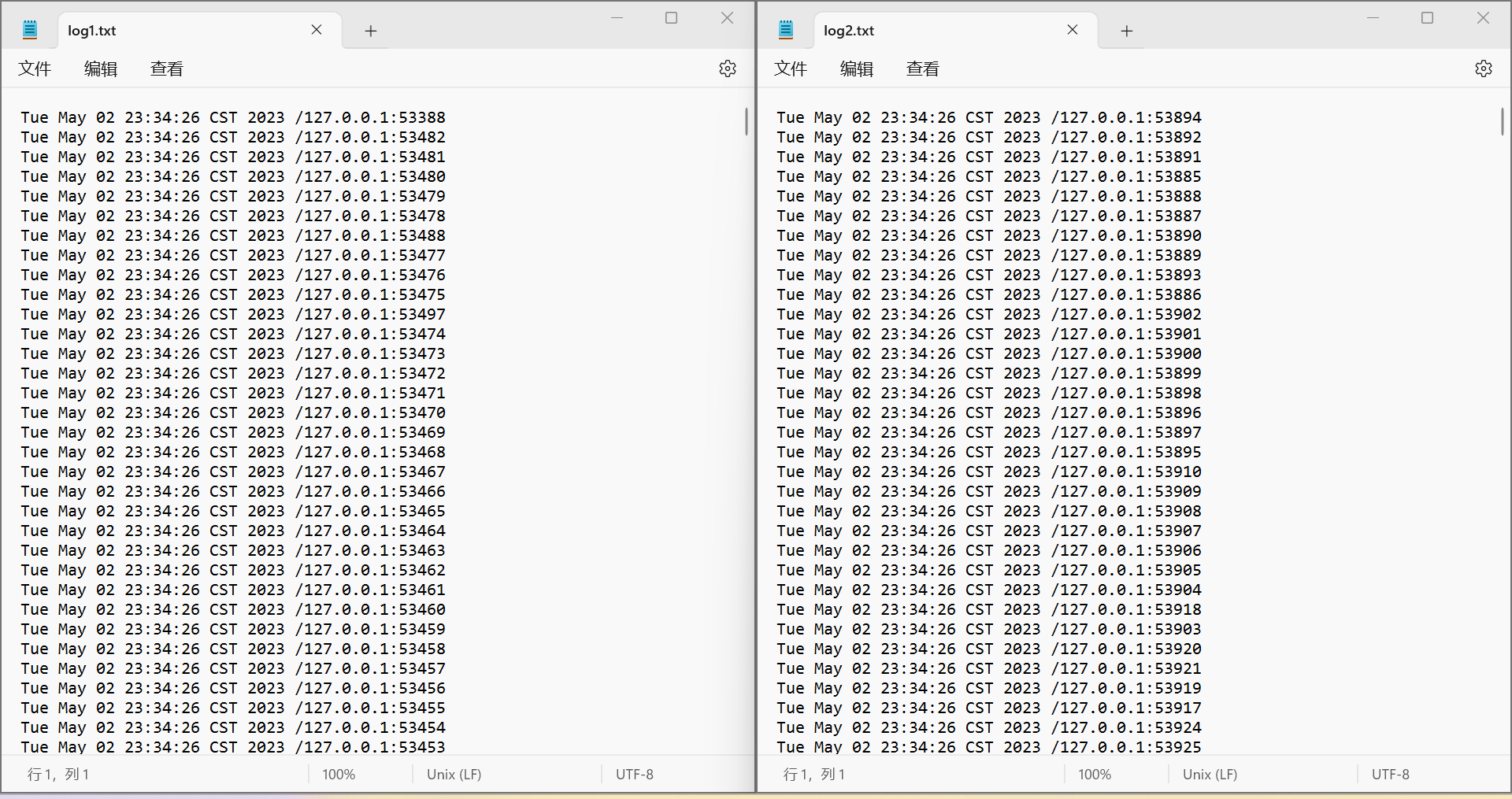
编写可重用的服务器日志程序模块，定义Logger类，记录客户端的IP地址和登录时间，根据传入的文件路径写入txt文件，由于多个线程可能会同时访问日志记录，所以使用synchronize同步锁修饰写入文件方法，保证每个线程对共享资源的访问按照一定顺序进行，保证线程安全性。MyLogger类代码如下：



Message为需要写入的信息，file为要写入的文件路径。这样，改变message和file即可实现重用。在两个服务器的run()方法中都加入如下代码：



两个日志的记录结果如下：



三、实验总结

本次实验，我编写了使用多线程处理客户端连接请求的TCP服务端程序、使用线程池处理客户端连接请求的TCP服务端程序、测试了两种方式的并发处理能力、编写了可重用的日志模块。

通过本次实验我回顾了线程的创建和开始方法、明白了线程池的工作方式以及线程池的创建和使用方式。接下来，我通过对多线程服务器和线程池服务器发起相同数目的连接请求次数，统计时间，得出线程池的运行时间小于多线程的运行时间，分析出线程池的并发处理能力高于多线程。因为多线程在大量线程的创建和销毁时要额外消耗时间，而线程池不用。最后，第一次学习什么是日志，了解到日志的作用和重要性，日志可以记录远程运行时的信息，在出错维护时起到重要作用。

在实验中也遇到了一些问题，在编写日志模块时不理解日志是什么东西，在两个服务器端分别写了两个日志，这样并没有实现可重用性。在发现实验要求需要可重用后，我分析两个的相同和不同之处，发现操作是相同的，只是写入的内容和写入的文件不同，因此，将文件和内容作为参数，得到一个新的类，类中方法实现写入即可。

总之，不是一次一帆风顺的实验，但是一次收获颇多的实验。

|  |
| --- |
| 指导教师批阅意见：  成绩评定：  指导教师签字：刘志丹  年 月 日 |
| 备注： |

注：1、报告内的项目或内容设置，可根据实际情况加以调整和补充。

2、教师批改学生实验报告时间应在学生提交实验报告时间后10日内。