**深 圳 大 学 课 程 作 业 报 告**

**课程名称： Java程序设计**

**课程作业名称： 基于大模型的在线代码审查助手**

**学院： 计算机与软件学院**

**专业： 人工智能卓越班**

**指导教师： 潘微科**

**报告人： 邓瑞霖 学号：2024150040 班级： 01**

**作业时间： 2025年11月17日（周一）~2025年12月24日（周三）**

**课程作业报告提交时间： 2025年12月16日星期二**

**教务部制**

|  |
| --- |
| **作业目的与要求：**  **课程作业目的：**Java程序设计综合练习，含多线程、I/O、网络通信等。  **课程作业要求：**  (1). 编写Java应用程序，实现“**基于大模型的在线代码审查助手**”。开发一个基于大模型的在线服务，能够并发地接收用户提交的Java代码片段，并返回由大模型生成的代码审查意见，例如，潜在缺陷、代码风格改进、性能优化建议等。   * **大模型：**成功集成一个常见的大模型API（例如：DeepSeek、阿里通义、百度文心、腾讯混元等）（20分） * **多线程&网络通信：**能够并行处理多个用户的网络请求，每个用户请求都应该在一个新的线程中处理，以避免阻塞主线程（20分） * **测试案例：**提供至少以下两个不同的Java代码片段作为测试案例（20分）   一个包含常见bug的代码片段  一个代码风格不佳的片段  一个存在性能隐患的片段   * **日志记录：**在本地文本文件（.txt）中记录所有代码审查活动（10分） * **特色功能：**可以列出自己实现的其他特色功能（每个特色功能5分，最多记两个特色功能共10分）   在报告中附上程序截图、完整的运行结果截图和详细的文字说明。  报告写作。要求：主要思路有明确的说明，重点代码有详细的注释，行文逻辑清晰可读性强，报告整体写作较为专业。（20分）  **说明：**  （1）本次课程作业满分为100分，占总成绩的比例（待定）。  （2）本次课程作业截至时间2025年12月24日（周三）21:59。  （3）报告正文：请在指定位置填写，本次课程作业**需要单独提交源程序文件**（源程序单独打包在Blackboard中上传，不要包含外部导入的包）。  （4）个人信息：WORD文件名中的“姓名”、“学号”，请改为你的姓名和学号；课程作业报告的首页，请准确填写“学院”、“专业”、“报告人”、“学号”、“班级”、“课程作业报告提交时间”等信息。  （5）提交方式：截至时间前，请在Blackboard平台中提交。  （6）发现抄袭（包括复制&粘贴整句话、整张图），**抄袭者和被抄袭者的成绩记零分。**  （7）延迟提交，不得分；如有特殊情况，请于截至日期之后的48小时内发邮件到panweike@szu.edu.cn，并在邮件中注明课程名称、作业名称、姓名、学号等信息，以及特殊情况的说明，我收到后会及时回复。  （8）期末考试阶段补交无效。 |

|  |
| --- |
| (1). 编写Java应用程序，实现“**基于大模型的在线代码审查助手**”。开发一个基于大模型的在线服务，能够并发地接收用户提交的Java代码片段，并返回由大模型生成的代码审查意见，例如，潜在缺陷、代码风格改进、性能优化建议等。   * **大模型：**成功集成一个常见的大模型API（例如：DeepSeek、阿里通义、百度文心、腾讯混元等）（20分） * **多线程&网络通信：**能够并行处理多个用户的网络请求，每个用户请求都应该在一个新的线程中处理，以避免阻塞主线程（20分） * **测试案例：**提供至少以下两个不同的Java代码片段作为测试案例（20分）   一个包含常见bug的代码片段  一个代码风格不佳的片段  一个存在性能隐患的片段   * **日志记录：**在本地文本文件（.txt）中记录所有代码审查活动（10分） * **特色功能：**可以列出自己实现的其他特色功能（每个特色功能5分，最多记两个特色功能共10分）   在报告中附上程序截图、完整的运行结果截图和详细的文字说明。  **项目核心思路与架构设计：**  采用客户端/服务端架构。服务器端负责接受客户端提交的代码并调用大模型API进行审查，并将结果返回给客户端  整体框架：  ·客户端（Client）：模拟用户，通过网络发送Java代码片段  ·服务端（Server）：  ·网络监听器：监听特定端口（8080），接受客户端连接  ·多线程池：管理工作线程，避免为每个请求创建新线程带来的开销  ·请求处理器：每个连接分配一个线程，独立处理：   1. 接受代码 2. 调用大模型API 3. 记录日志 4. 发送审查结果   ·日志模块：负责将活动记录到本地文件(review\_log.txt)  ·大模型API集成模块：封装与外部大模型的HTTP通信逻辑  代码实现思路：   1. 服务器端：多线程与网路通信（CodeReviewServer.java文件)   使用ServerSocket循环监听连接，并将每个新的客户端连接交给一个线程池处理，并实现并发性   1. 请求处理器（ClientHandler.java文件）   每个ClienHandler实现Runnable接口，负责完整的代码审查流程   1. 大模型集成（LLMService.java文件）   选择通义大模型，模型选择为qwen-plus  现在官网中获取到通义大模型的API Key  思路：构造包含Java代码和审查指令的json请求体，通过HTTP POST发送给大模型API接口，并解析返回的json结果  代码：   1. 日志记录（Logger.java文件）   使用该FileWriter以追加模式将所有审查活动记录到本地文件review\_log.txt   1. 案例代码（CodeRevierClient.java文件）   负责连接服务器并发送测试代码，将要测试的代码放到一个数组中，三个样例子：分别为数组越界，代码风格不佳，存在性能隐患三段代码  三个样例：  将测试样例提交到大模型中：  特色功能：   1. 实现并发测试：启动5个客户端同时提交代码：  1. 实现代码提交历史记录保存   除了保存记录日志外，将客户端提交的代码片段单独存储在/history目录下，并以时间来命名（如20251215-152500-clientIP.java），方便后续的人工审查和追踪  在ClientHandler.java中：  首先定义保存的路径，src/Class/Homework/history  保存代码的方法：  在run方法中：  运行过程：  打开两个终端，分别为客户端和服务端：  编译所有java文件，将文件放到src/Class/Homework下  编译所有文件并放到bin目录下：javac -d bin -cp "src" src/Class/Homework/\*.java  在其中一个终端中运行：java -cp "bin" Class.Homework.CodeReviewServer  在另一个终端中运行：java -cp "bin" Class.Homework.CodeReviewClient  运行CodeReviewCient会提交待审查的代码，并将审查的结果保存在review\_log.txt文件中，运行得到的日志：  例如：案例1:存在数组越界  案例2:代码风格不佳  案例三：  并发性测试，采用多个客户端提交同一段代码：  记录的历史代码片段：  ++++++++++++++++++++++++++++++++++++++++++++++++++++++  **其他（例如感想、建议等等）。**  **技术实践上的收获：**   * 通过亲手实现一个高并发服务器，我深刻理解了阻塞I/O模型的工作机制、线程池在避免频繁创建销毁线程开销上的优势，以及如何通过Synchronized关键字或ExecutorService进行线程同步与管理。 * 从阅读官方文档、申请API Key，到使用HttpURLConnection构建HTTP请求、处理JSON格式的请求与响应，最后实现异常重试机制，这一整套流程让我掌握了现代软件开发中微服务集成的基本方法论，认识到清晰接口文档的重要性。 * 我学会了运用职责分离的原则来设计系统。将服务器(CodeReviewServer)、请求处理器(ClientHandler)、大模型服务(LLMService)、日志模块(Logger)等模块解耦，使得代码结构清晰、易于维护和扩展。   **遇到的问题与解决方案：**   * 挑战一：最初使用简单的字符串匹配来提取审查结果，但当响应内容包含转义字符（如\n、\"）或格式稍有变化时，解析极易失败。解决方案是采用更健壮的Pattern和Matcher进行正则表达式匹配，并编写了专门的parseResponse方法处理转义字符，大大提升了程序的稳定性。 * 挑战二：多个线程同时写入同一日志文件可能导致内容错乱。我通过将日志操作封装成类，并利用FileWriter的追加模式和恰当的同步控制，确保了日志信息的完整性和时序性，为问题追溯提供了可靠依据。   **对代码审查工作的新认识：**  在设计和测试案例的过程中，我对代码质量有了更深的理解。不仅仅是寻找NullPointerException或语法错误，优秀的代码审查更关注：   * 可读性：清晰的命名、合理的注释、一致的代码风格是团队协作的基石。 * 健壮性：对边界条件（如空数组、非法输入）的充分处理。 * 性能：避免在循环内进行字符串拼接等低效操作。 * 设计模式：如何通过更好的设计使代码更灵活、更易扩展。   本项目通过大模型自动化了基础审查，但我认识到，高级的架构设计和业务逻辑合理性审查仍需经验丰富的人工介入，人机协同将是未来软件开发的重要模式。 |

深圳大学学生课程作业报告用纸

|  |
| --- |
| 指导教师批阅意见：  成绩评定：  指导教师签字：  2025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、教师批改学生课程作业报告时间应在学生提交课程作业报告时间后10日内。