**深 圳 大 学 实 验 报 告**

**课程名称： Java 程序设计**

**实验项目名称： 实验4 I/O、GUI和网络编程**

**学院： 计算机与软件学院**

**专业： 计算机科学与技术**

**指导教师： 姚俊梅**

**报告人： 杜良衡 学号： 2022150255 班级： 01B**

**实验时间：2024年11月26日 ~ 2024年12月28日**

**实验报告提交时间： 2024.12.18**

**教务部制**

|  |
| --- |
| **实验目的与要求：**  **实验目的：**掌握Java程序设计开发环境的搭建，掌握编译、运行等基本步骤和命令；掌握基础知识和基本类型应用，以及类的初级应用。  **实验要求：**  1、输入输出应用（共1题，30分）  (1) 编写一个 Java 程序，自行从BB上下载日志文件logs.txt，其中每行的格式为：[时间戳] [日志级别] [消息内容]。请实现以下功能：  i) 判断日志文件是否存在。如果不存在，提示用户文件缺失并退出程序。  ii) 统计每种日志级别（INFO、WARN、ERROR）的数量，并将结果输出到控制台。  iii) 按日志级别分类，将 ERROR 日志写入 error\_logs.txt，WARN 日志写入 warn\_logs.txt，其余日志写入 info\_logs.txt。每个日志都按照时间先后顺序排序。   1. 在完成上述任务后，读取分类后的文件，打印文件内容到控制台。   2、GUI应用（共1题，30分）  (1) 编写Java应用程序，实现“Java机考”的功能（“单机版Java简易机考程序”），包含单选题、多选题和判断题三种题型。  在主线程中创建一个Frame类型的窗口，在该窗口中再创建一个线程giveQuestion。  线程giveQuestion每隔20秒钟输出一个选择题（含A,B,C,D共4个选项，要求支持单选题和多选题，单选题用radio button，多选题用check box）或一个判断题（用radio button），选择题和判断题混合着给出；用户输入答案并按提交按钮提交结果（达到20秒自动提交结果）；程序判断用户输入的答案是否正确（如果错选或漏选均得零分），并实时显示当前题目的正确答案、已经给出的题目的数量（分别给出单选题数量、多选题数量和判断题数量）、用户答对的数量（分别给出单选题数量、多选题数量和判断题数量）、用户的成绩和用户答题所花的总的时间。  如此循环15次，包括随机选择的5个单选题（每题1分）、随机选择的5个多选题（每题2分）和随机选择的5个判断题（每题1分），结束测试时给出最终成绩。  题库应包含10个单选题、10个多选题和10个判断题。要求使用图形用户界面。  3、网络编程（30分）  (1) 请编写JAVA程序，实现一个线上聊天系统。要求一个用户发消息，所有用户都能收到，即一个客户端把聊天消息发给服务器，服务器再将收到的消息转发给所有客户端。需使用TCP通信模式，并实现GUI界面。具体要求如下：  (i) 服务器端Server类   * 为了同时添加多个用户参与聊天，需使用多线程添加用户。 * 在收到一个用户发送的消息后，需转发给此用户外的所有其他用户。 * 如收到“exit”代表用户退出，需删除此用户。   (ii) 客户端Client类   * 可以主动发送消息，如果想退出可以给服务器发送“exit”。 * 需以 “用户名：消息” 的形式显示所有本用户发送和接收的消息。   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。（10）  **说明：**  （1）本次实验课作业满分为100分。  （2）报告正文：请在指定位置填写，本次实验不需要单独提交源程序文件。  （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；实验报告的首页，请准确填写“学院”、“专业”、“报告人”、“学号”、“班级”、“实验报告提交时间”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（含抄袭往届同学的作业）。**  （7）延迟提交将酌情扣分；期末考试阶段补交无效。  **1、输入输出应用（共1题，30分）**  (1) 编写一个 Java 程序，自行从BB上下载日志文件logs.txt，其中每行的格式为：[时间戳] [日志级别] [消息内容]。请实现以下功能：  i) 判断日志文件是否存在。如果不存在，提示用户文件缺失并退出程序。  ii) 统计每种日志级别（INFO、WARN、ERROR）的数量，并将结果输出到控制台。  iii) 按日志级别分类，将 ERROR 日志写入 error\_logs.txt，WARN 日志写入 warn\_logs.txt，其余日志写入 info\_logs.txt。每个日志都按照时间先后顺序排序。   1. 在完成上述任务后，读取分类后的文件，打印文件内容到控制台。   首先将下载的logs.txt文件放到项目的根目录下，并在该位置创建error\_logs.txt、warn\_logs.txt和info\_logs.txt分别来存储不同日志级别的日志。  创建LogProcessor类来进行日志处理。先在main方法中定义源日志文件log.txt和按日志级别分类的目标日志文件的路径，并使用Files.exists方法检查源日志文件是否存在于指定路径。如果文件不存在，则打印错误信息并退出程序。    接着创建HashMap来存储不同级别的日志条目，为日志级别名作为键，为每种级别初始化一个空列表。  为了便于逐行读取文件，同时保证读取的高效率，使用BufferedReader来读取源日志文件内容。该对象会通过readLine方法遍历文件的每一行，通过正则表达式来匹配日志行中的时间戳、日志级别和消息内容并进行提取。然后根据提取的日志级别将日志条目添加到相应的列表中。如果遇到无法解析的日志行，则输出警告信息。因为这里涉及到文件的读取，所以需要用try-catch块来捕获并处理可能发生的IO异常。  文件解析完成后，会遍历logsByLevel映射，对每个日志级别的列表进行排序，确保日志按照时间先后顺序排列。然后调用writeLogsToFile方法，将分类后的日志条目分别写入到对应的文件中。  将日志列表写入指定文件的writeLogsToFile方法以写入文件路径和日志条目列表作为参数，同时为了提高写入效率，使用了BufferedWriter来遍历日志列表并逐行写入文件。  写入操作完成后，主方法会计算并打印出每种日志级别的数量来作为日志统计信息。最后调用printFileContents方法打印分类后文件的内容。  printFileContents方法中以文件路径作为参数，同样会使用BufferedReader来读取文件所有行，并将其逐行打印到控制台。  运行程序，控制台成功输出每种日志级别数量的统计结果。  不同级别的日志写入到了对应的文件中，并按照时间先后顺序排序。  分类后的文件内容也被成功读取并打印到了控制台。  **2、GUI应用（共1题，30分）**  (1) 编写Java应用程序，实现“Java机考”的功能（“单机版Java简易机考程序”），包含单选题、多选题和判断题三种题型。  在主线程中创建一个Frame类型的窗口，在该窗口中再创建一个线程giveQuestion。  线程giveQuestion每隔20秒钟输出一个选择题（含A,B,C,D共4个选项，要求支持单选题和多选题，单选题用radio button，多选题用check box）或一个判断题（用radio button），选择题和判断题混合着给出；用户输入答案并按提交按钮提交结果（达到20秒自动提交结果）；程序判断用户输入的答案是否正确（如果错选或漏选均得零分），并实时显示当前题目的正确答案、已经给出的题目的数量（分别给出单选题数量、多选题数量和判断题数量）、用户答对的数量（分别给出单选题数量、多选题数量和判断题数量）、用户的成绩和用户答题所花的总的时间。  如此循环15次，包括随机选择的5个单选题（每题1分）、随机选择的5个多选题（每题2分）和随机选择的5个判断题（每题1分），结束测试时给出最终成绩。  题库应包含10个单选题、10个多选题和10个判断题。要求使用图形用户界面。  创建继承JFrame类的JavaExamination类来构建Java机考程序的图形化界面。在该类中首先定义各种常量和全局变量，包括各题型的数量及其对应的分数、题目间的时间间隔（此处为20秒）、测试统计信息（总分数、各题型已答题目数、正确题目数等）和GUI基本组件。  接着初始化题库题目列表、当前测试题目列表以及当前显示的题目。Question是一个抽象类，负责定义所有题型的共有属性和方法。  JavaExamination类的构造函数用于初始化整个程序。首先会初始化主窗口，设置标题、窗口大小等，接着分别调用initializeAllQuestions和selectRandomQuestions方法来初始化题库以及从题库中随机选择一定数量的题目作为测试题。  然后设置GUI组件，包括问题面板、提交按钮和状态面板。提交按钮添加了SubmitActionListener监听器，用于点击该按钮时提交答案并显示下一题。状态面板中包含了答题时间、正确答案、用户当前的分数以及已答题目统计信息。  初始化图形界面完成后会利用startTime记录当前时间作为测试开始时间，并调用startTimer方法来启动定时器，用于时间到20秒后自动提交答案。最后调用giveNextQuestion方法给出第一道题。  SubmitActionListener类实现了ActionListener接口，负责监听和处理提交按钮的点击事件。当用户点击了提交答案按钮，就会调用submitAnswer方法来提交当前答案，并调用giveNextQuestion方法显示下一题。  在初始化题库中所有题目的initializeAllQuestions方法中，会先将前面定义的allQuestions列表初始化为ArrayList数据列表，以便后面进行随机访问的操作。接着分别添加10个单选题、10个多选题和10个判断题到该列表，添加时会以题型名称加编号来作为每个题目的唯一标识。  用于从题库中随机抽取指定数量题目的selectRandomQuestions方法会从allQuestions列表中随机选择5个单选题、5个多选题和5个判断题并添加到questions列表中作为用户的测试题。为了增加测试题目的不确定性，会调用Collections类的静态方法shuffle来打乱该列表中题目的顺序。  用于显示下一道题的giveNextQuestion方法会先判断当前测试题列表是否为空，是的话说明没有更多题目，调用endExam方法结束测试并返回。还有题目的话则会从列表中取出该题目，将程序窗口问题面板中原有的信息清空，并更新该题目信息到面板。  startTimer方法用于在测试开始时设置一个定时器，该定时器会在指定的时间间隔后，此处为题目之间的时间间隔20秒，触发一次事件。当事件被触发时，会自动调用submitAnswer方法提交当前题目的答案，并调用giveNextQuestion方法进入下一道题。  submitAnswer方法用于提交用户答案并检查其正确性。该方法首先会根据题型更新在状态面板中呈现的已经给出题目数量。接着问题对象会调用对应的checkAnswer方法判断答案是否正确。如果正确则会根据题型更新状态面板中已答对题目数量。最后会调用updateLabels方法更新程序面板中的分数和时间，并调用displayCorrectAnswer方法显示正确答案。  updateLabels方法负责更新状态面板中用户当前各题型的答题数和正确数，并以当前时间与测试开始时间startTime的差作为用户作答时间更新面板中对应部分。  displayCorrectAnswer方法用于显示上一题的正确答案，会在用户点击提交按钮或到时间自动提交的时候调用。  endExam方法会在用户答题完成后调用，该方法会停止定时器，并显示用户的最终分数和总用时。最后会通过dispose方法来关闭并释放主窗口资源。  Question抽象类定义了所有题型的共有属性和方法。其属性包括题目文本和题目对应的分数。题目文本text在前面已经介绍过，是由题型与编号所拼接，作为每个题目的唯一标识。该类的有参构造器会初始化这两个属性。该抽象类的方法包括获取显示问题的JPanel的getPanel方法，检查答案是否正确的checkAnswer方法，获取正确答案文本的getCorrectAnswerText方法以及返回题目分数的getScore方法。  SingleChoiceQuestion类用于表示单选题。该类的成员变量包括选项列表options、正确答案选项correctOption以及用户选择的选项selectedOption，同时还包括了父类Question类的属性text和score。  该类实现了父类Question类中声明的三个抽象方法。getPanel方法用于创建并返回一个JPanel，该面板包括了单选题的题目描述和所有选项的单选按钮，即JRadioButton。用户可以通过点击这些按钮来选择答案。  checkAnswer方法会判断用户选择的答案是否与正确答案匹配，getCorrectAnswerText方法会直接返回正确答案的文本。  MultiChoiceQuestion类用来表示多选题，其成员变量和方法与SingleChoiceQuestion类相似。  表示判断题的TrueFalseQuestion类的成员变量有所不同，表示正确答案的correctAnswer为boolean类型，还包含了True和False选项按钮。其方法则与前两个类相似。  最后在main方法中，会先使用SwingUtilities.invokeLater方法确保所有Swing组件的操作都在正确的线程中执行，接着调用Lambda表达式创建JavaExamination实例，并通过setVisible(true)使创建的程序窗口可见。  编写代码完成后启动程序，可以看到此时成功呈现程序的图形化界面。  在该界面停留20秒后，程序自动提交了答案。此时状态面板中显示了用户当前的分数与答题情况，作答时间进行了更新，面板也显示了上一题的正确答案。  可以看到不同题型的选项都可以正确显示，而答对后的加分也与要求一致。  在完成15题后，各题型随机选取的数量均符合预期，且此时界面成功呈现了用户最终的分数与总用时，点击确定即可退出程序。  **3、网络编程（30分）**  (1) 请编写JAVA程序，实现一个线上聊天系统。要求一个用户发消息，所有用户都能收到，即一个客户端把聊天消息发给服务器，服务器再将收到的消息转发给所有客户端。需使用TCP通信模式，并实现GUI界面。具体要求如下：  (i) 服务器端Server类   * 为了同时添加多个用户参与聊天，需使用多线程添加用户。 * 在收到一个用户发送的消息后，需转发给此用户外的所有其他用户。 * 如收到“exit”代表用户退出，需删除此用户。   (ii) 客户端Client类   * 可以主动发送消息，如果想退出可以给服务器发送“exit”。 * 需以 “用户名：消息” 的形式显示所有本用户发送和接收的消息。   先编写服务器端Server类，首先定义端口常量PORT，指定服务器监听的TCP端口号为12345，接着创建同步集合clients用来保存所有已连接客户端的输出流，即PrintWriter对象，从而确保多线程环境下对集合的操作是线程安全的。  接着在main方法中编写服务器启动逻辑，使用try-with-resources语法创建一个ServerSocket实例并绑定到指定端口，便于最后资源的释放。服务器启动成功后会输出提示信息，并进入while循环，调用accept方法等待新的客户端连接。每当有一个新连接到达时，会创建一个新的ClientHandler线程来处理该连接，并调用start方法立即开始执行。  ClientHandler内部类继承自Thread，每个实例代表一个独立的线程，用于处理单个客户端的通信。该类的成员变量包含客户端的Socket对象，用于写入数据的PrintWriter对以及用于读取数据的BufferedReader对象。该类还包括一个有参构造器，用于初始化客户端套接字。  在重写的run方法中，定义了通信线程的具体任务。先初始化输入输出流，并将新客户端的输出流添加到全局集合clients中。接着循环读取来自客户端的消息，如果消息不是“exit”，则将其广播给所有其他客户端。  如果出现IO异常或收到“exit”消息，都会在finally块中执行用户退出操作：从集合中移除对应的输出流，并关闭相关资源。  接着来编写客户端类。该类先定义了成员变量，包括构成GUI的各种组件以及用于网络通信的对象。在其构造函数中，接收用户名作为参数，并调用initializeUI方法设置GUI。  在初始化用户界面的initializeUI方法中，先创建了主窗口，并设置了JFrame的基本属性，例如标题、关闭操作、大小等。接着使用不可编辑的JTextArea来显示聊天内容，并将其包裹在JScrollPane中以便滚动查看长消息。  然后创建输入面板，包括一个文本字段用于输入消息和一个按钮用于发送消息，两者都被放置在一个JPanel中，并添加到窗口底部。最后利用setVisible方法使窗口可见，并调用connectToServer方法尝试连接到服务器。  在连接服务器的connectToServer方法中，会尝试通过指定的主机名（此处为localhost）和端口号创建一个Socket对象。然后获取输入输出流，准备读取来自服务器的消息或向服务器发送消息。最后创建并启动一个新的ReceiveMessageThread线程来监听服务器发来的消息。  发送按钮监听器SendButtonListener类实现了ActionListener接口，用于处理消息发送逻辑。该事件触发后会从文本字段获取用户输入的消息，如果是“exit”，则关闭连接并退出程序；否则将消息格式化后发送给服务器，并清空文本字段。  接收消息线程ReceiveMessageThread继承自Thread，负责持续监听来自服务器的消息。一旦接收到新消息，就会通过chatArea.append将其追加到聊天区域中供用户查看。  主方法负责启动客户端，使用SwingUtilities.invokeLater确保GUI相关代码在AWT事件调度线程上执行，避免并发问题，并创建一个新的客户端实例，赋予其随机生成的用户名。  编写完成后进行测试，先启动服务器端。  再启动两个客户端，成功显示图形化用户界面。  可以看到两个客户端都能成功发送和接收消息。  当客户端Client 1发送exit消息后，用户成功退出。 |

|  |
| --- |
| ++++++++++++++++++++++++++++++++++++++++++++++++++++++  **其他（例如感想、建议等等）。**  在本次实验过程中，我对File类和输入输出流、图形用户界面设计以及Java网络编程有了更加深入的了解：通过文件读取与解析的练习，掌握了如何利用BufferedReader实现高效的字符输入流读取；通过机考程序的编写，熟悉了构建图形用户界面的基本步骤，还有AWT和Swing组件的使用；通过线上聊天室系统的实现，明白了服务端与客户端之间是怎样进行消息的发送与接收。在编写程序时，也遇到了不少问题，例如对GUI的各种组件不是很熟悉，导致界面总是不能呈现出预期的效果。不过从网上搜索相关教程，并进行更加深入的学习后，这些问题也能够迎刃而解。 |

深圳大学学生实验报告用纸

|  |
| --- |
| 指导教师批阅意见：  成绩评定：  指导教师签字：  2024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。