**深 圳 大 学 实 验 报 告**

**课程名称：Java 程序设计**

**实验项目名称：课程实验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：潘微科**

**报告人：曹秦鲁 学号： 2023150203 班级：国际班**

**实验时间：2024年11月29日（周五）-2024年12月18日（周三）**

**实验报告提交时间：**

**教务部制**

|  |
| --- |
| **实验目的与要求：**  **实验目的：**掌握I/O程序设计，能够读写本地文件等，初步掌握图形界面程序设计，掌握网络通信协议及相关程序设计。  **实验要求：**  **Part 1（25分）**  (1.1). 数据解析和统计。<https://snap.stanford.edu/data/web-Amazon.html> 网站上有很多Amazon的数据集供研究人员下载使用。本次实验使用Watches.txt.gz数据集，请下载后解压。格式说明请看网页上的“Data Format”部分。在报告中附上程序截图、运行结果（如每个输出文件前10行的截图等）和简要文字说明。  （i）使用Java语言读取解压后的文件（Watches.txt），并得到以下文件（10分）：  **review.txt**：每行2列，以分号作为分隔符，第1列是userID，第2列是productID，表示(user, product)二元组。该文件中不同行之间的顺序，按照userID从小到大排列，当userID相同时按照productID从小到大排列。  注：删除ID为unknown的记录；排序时按照字符串顺序。  （ii）使用Java语言根据review.txt进行计算，并得到以下文件（15分）：  **productNeighborhood.txt**：每行4列，以分号作为分隔符，第1列是productID，第2-4列是与该product最相似（根据相似度值）的3个product的productID，按相似度值从大到小排列（当相似度相同时，按照productID的字符串顺序从小到大排序），其中相似度是通过review.txt中的两列的信息计算得到的Jaccard index值。该文件中不同行之间的顺序，按照第1列的productID从小到大排列。关于相似度的计算，要求使用多线程实现（5分）。  注：计算两个商品（product）之间的相似度（即Jaccard index）时，根据这两个商品所关联的用户的集合来计算——集合的交集的大小除以集合的并集的大小。  **Part 2（25分）**  (2.1).使用JLabel、JTextArea、JButton等控件实现句子中英互译的demo，该demo包含两个文本框，第一个文本框用于输入中文句子或显示第二个文本框中的英文句子的中文翻译，第二个文本框用于输入英文句子或显示第一个文本框中的中文句子的英文翻译。每个文本框下方各有一个按钮，第一个按钮的名称是“中译英”，第二个按钮的名称是“英译中”，点击按钮表示将该文本框中的内容翻译成另一种语言。要求使用以下三种方式：  A、使用自己事先准备好的中英文翻译（此部分占5分）；  B、使用百度翻译API、有道翻译API或其他API中的一个API（此部分占5分）；  C、使用腾讯混元大模型、百度文心一言大模型API、阿里通义千问大模型API或其他大模型API中的两个API（此部分占10分）；  要求使用以下两个例句：  建校41年，深圳大学秉承“自立、自律、自强”的校训，紧随特区，锐意改革、快速发展，为特区发展和国家现代化建设做出了重要贡献。  Sticking to the motto of “self-reliance, self-discipline, self-improvement”, the University is dedicated to serving the Shenzhen Special Economic Zone (SEZ), demonstrating China’s reform and opening up and pioneering change in higher education.  要求使用图形用户界面，界面美观、交互友好。在报告中附上程序截图、运行结果和详细的文字说明。（5分）  **Part 3（30分）**  (3.1). 利用套接字连接（TCP）编写程序，该程序包括三个客户端（ClientA、ClientB、ClientC）和一个服务端（ServerS），三个客户端通过服务端作为桥梁实现相互间的文字交流，例如，ClientA先发信息给ServerS，然后ServerC再将收到的信息转发给ClientB和ClientC。在报告中附上示意图（三个客户端+一个服务端）、程序截图、完整的运行结果和简要文字说明。（20分）  (3.2). 利用数据报通信（UDP）实现题(1)中的要求。（10分）  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）本次实验课作业满分为100分，占总成绩的比例7%。  （2）本次实验课作业截至时间2024年12月18日（周三）21:59。  （3）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。**  （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。 |

|  |
| --- |
| **Part 1（25分）**  (1.1). 数据解析和统计。<https://snap.stanford.edu/data/web-Amazon.html> 网站上有很多Amazon的数据集供研究人员下载使用。本次实验使用Watches.txt.gz数据集，请下载后解压。格式说明请看网页上的“Data Format”部分。在报告中附上程序截图、运行结果（如每个输出文件前10行的截图等）和简要文字说明。  （i）使用Java语言读取解压后的文件（Watches.txt），并得到以下文件（10分）：  **review.txt**：每行2列，以分号作为分隔符，第1列是userID，第2列是productID，表示(user, product)二元组。该文件中不同行之间的顺序，按照userID从小到大排列，当userID相同时按照productID从小到大排列。  注：删除ID为unknown的记录；排序时按照字符串顺序。r  一、项目概述  本项目实现了Amazon手表评论数据的处理功能，从原始评论数据中提取用户ID和产品ID，并按照特定规则排序生成新的数据文件。  二、代码实现分析  1. 主类结构  主类*`AmazonProductSimilarityAnalyzer`*的核心实现：  *```java*  *public class AmazonProductSimilarityAnalyzer {*  *public static void main(String[]* args*) {*  *try (BufferedReader* *reader* *= new BufferedReader(new FileReader(*"Watches.txt"*))) {*  *Set*<*Review*> *reviews* *= new TreeSet<>(); // 使用TreeSet自动排序*    *String* *line;*  *String* *currentUserId* *= null; // 临时存储当前处理的userId*  *String* *currentProductId* *= null; // 临时存储当前处理的productId*    *while ((line = reader.readLine()) != null) {*  *if (line.trim().isEmpty()) {*  *if (currentUserId != null && currentProductId != null*  *&& !currentUserId.equals(*"unknown"*)) {*  *reviews.add(new Review(currentUserId, currentProductId));*  *}*  *currentUserId = null;*  *currentProductId = null;*  *continue;*  *}*    *String*[] *parts* *= line.split(*": "*, 2);*  *if (parts.length != 2) continue;*    *String* *key* *= parts[0].trim();*  *String* *value* *= parts[1].trim();*    *switch (key) {*  *case* "review/userId"*:*  *currentUserId = value;*  *break;*  *case* "product/productId"*:*  *currentProductId = value;*  *break;*  *}*  *}*    *try (PrintWriter* *reviewWriter* *= new PrintWriter(*"reviews.txt"*)) {*  *for (Review* *review* *: reviews) {*  *reviewWriter.println(review.userId +* ";" *+ review.productId);*  *}*  *}*  *}*  *}*  *}*  *```*  主要功能：  - 使用BufferedReader读取输入文件  - 使用TreeSet实现自动排序  - 使用PrintWriter输出结果  2. 数据结构设计  2.1 Review类  内部类Review的实现：  *```java*  *private static class Review implements* Comparable*<Review> {*  *String* *userId;*  *String* *productId;*    *@Override*  *public int compareTo(Review* other*) {*  *// 首先按userId排序*  *int* *userCompare* *= this.userId.compareTo(other.userId);*  *if (userCompare != 0) {*  *return userCompare;*  *}*  *// userId相同时按productId排序*  *return this.productId.compareTo(other.productId);*  *}*  *}*  *```*  关键点说明：  - 实现Comparable接口以支持排序  - compareTo方法实现两级排序：  1. 首先按userId从小到大排序  2. userId相同时按productId排序  3. 核心功能实现  3.1 文件读取与解析  *```java*  *while ((line = reader.readLine()) != null) {*  *if (line.trim().isEmpty()) {*  *// 遇到空行时处理当前记录*  *if (currentUserId != null && currentProductId != null*  *&& !currentUserId.equals(*"unknown"*)) {*  *reviews.add(new Review(currentUserId, currentProductId));*  *}*  *// 重置临时变量*  *currentUserId = null;*  *currentProductId = null;*  *continue;*  *}*    *String*[] *parts* *= line.split(*": "*, 2);*  *if (parts.length != 2) continue;*    *String* *key* *= parts[0].trim();*  *String* *value* *= parts[1].trim();*    *switch (key) {*  *case* "review/userId"*:*  *currentUserId = value;*  *break;*  *case* "product/productId"*:*  *currentProductId = value;*  *break;*  *}*  *}*  *```*  代码说明：  1. 文件读取逻辑：  - 按行读取文件  - 使用空行作为记录分隔符  - 使用*`: `*分割键值对  2. 数据处理逻辑：  - 提取userId和productId  - 跳过userId为unknown的记录  - 确保数据完整性  3. 数据存储：  - 使用TreeSet自动维护排序  - 每条记录包含userId和productId  - 按指定规则排序存储  三、性能分析  1. 时间复杂度分析：  - 文件读取：O(n)，n为文件行数  - TreeSet添加操作：O(log m)，m为有效评论数  - 总体：O(n + m log m)  2. 空间复杂度分析：  - TreeSet存储：O(m)，m为有效评论数  - 临时变量：O(1)  - 总体：O(m)  运行结果：  （ii）使用Java语言根据review.txt进行计算，并得到以下文件（15分）：  **productNeighborhood.txt**：每行4列，以分号作为分隔符，第1列是productID，第2-4列是与该product最相似（根据相似度值）的3个product的productID，按相似度值从大到小排列（当相似度相同时，按照productID的字符串顺序从小到大排序），其中相似度是通过review.txt中的两列的信息计算得到的Jaccard index值。该文件中不同行之间的顺序，按照第1列的productID从小到大排列。关于相似度的计算，要求使用多线程实现（5分）。  注：计算两个商品（product）之间的相似度（即Jaccard index）时，根据这两个商品所关联的用户的集合来计算——集合的交集的大小除以集合的并集的大小。  1. 数据读取与预处理  *```java*  *// 存储商品-用户映射关系*  *private static Map*<*String*, *Set*<*String*>> *productToUsers* *= new HashMap<>();*  *// 从reviews.txt读取数据，构建商品-用户映射*  *private static void buildProductToUsers(String inputFile) throws IOException {*  *try (BufferedReader* *br* *= new BufferedReader(new FileReader(inputFile))) {*  *String* *line;*  *int* *count* *= 0;*  *while ((line = br.readLine()) != null && count < MAX\_LINES) {*  *String*[] *parts* *= line.split(*";"*);*  *if (parts.length != 2) continue;*  *String* *user* *= parts[0].trim();*  *String* *product* *= parts[1].trim();*  *productToUsers.computeIfAbsent(product, k -> new HashSet<>()).add(user);*  *count++;*  *}*  *}*  *}*  *```*  2. 相似度计算  - 采用Jaccard相似度计算公式：|A∩B| / |A∪B|  - 使用多线程并行计算所有商品对的相似度  *```java*  *// 计算两个商品的Jaccard相似度*  *int* *intersectionSize* *= 0;*  *for (String* *u* *: users1) {*  *if (users2.contains(u)) {*  *intersectionSize++;*  *}*  *}*  *int* *unionSize* *= users1.size() + users2.size() - intersectionSize;*  *double* *similarity* *= unionSize == 0 ? 0.0 : ((double) intersectionSize / unionSize);*  *```*  3. 并行计算框架  *```java*  *// 创建线程池*  *int* *numThreads* *= Runtime.getRuntime().availableProcessors();*  *ExecutorService* *executor* *= Executors.newFixedThreadPool(numThreads);*  *// 提交计算任务*  *for (int* *i* *= 0; i < products.size(); i++) {*  *for (int* *j* *= i + 1; j < products.size(); j++) {*  *String* *p1* *= products.get(i);*  *String* *p2* *= products.get(j);*  *futures.add(executor.submit(new ComputeTask(p1, p2)));*  *}*  *}*  *```*  4. 结果处理  1. 收集每个商品的相似度结果  *```java*  *Map*<*String*, *List*<*SimilarityResult*>> *productSimilarities* *= new HashMap<>();*  *for (Future*<*List*<*SimilarityResult*>> *f* *: futures) {*  *List*<*SimilarityResult*> *resultList* *= f.get();*  *for (SimilarityResult* *r* *: resultList) {*  *productSimilarities.get(r.product1).add(new SimilarityResult(r.product1, r.product2, r.similarity));*  *productSimilarities.get(r.product2).add(new SimilarityResult(r.product2, r.product1, r.similarity));*  *}*  *}*  *```*  2. 排序并选择前三个最相似的商品  *```java*  *Collections.sort(list, new Comparator<SimilarityResult>() {*  *@Override*  *public int compare(SimilarityResult* o1*, SimilarityResult* o2*) {*  *int* *cmp* *= Double.compare(o2.similarity, o1.similarity);*  *if (cmp != 0) {*  *return cmp;*  *} else {*  *return o1.product2.compareTo(o2.product2);*  *}*  *}*  *});*  *Collections.reverse(list); // 确保从大到小排序*  *```*  5. 输出结果  - 将每个商品的前三个最相似商品写入文件  - 格式：商品ID;相似商品1;相似商品2;相似商品3  *```java*  *writer.write(p +* ";" *+ top3.get(0) +* ";" *+ top3.get(1) +* ";" *+ top3.get(2));*  *```*  运行结果：  **Part 2（25分）**  (2.1).使用JLabel、JTextArea、JButton等控件实现句子中英互译的demo，该demo包含两个文本框，第一个文本框用于输入中文句子或显示第二个文本框中的英文句子的中文翻译，第二个文本框用于输入英文句子或显示第一个文本框中的中文句子的英文翻译。每个文本框下方各有一个按钮，第一个按钮的名称是“中译英”，第二个按钮的名称是“英译中”，点击按钮表示将该文本框中的内容翻译成另一种语言。要求使用以下三种方式：  A、使用自己事先准备好的中英文翻译（此部分占5分）；  B、使用百度翻译API、有道翻译API或其他API中的一个API（此部分占5分）；  C、使用腾讯混元大模型、百度文心一言大模型API、阿里通义千问大模型API或其他大模型API中的两个API（此部分占10分）；  要求使用以下两个例句：  建校41年，深圳大学秉承“自立、自律、自强”的校训，紧随特区，锐意改革、快速发展，为特区发展和国家现代化建设做出了重要贡献。  Sticking to the motto of “self-reliance, self-discipline, self-improvement”, the University is dedicated to serving the Shenzhen Special Economic Zone (SEZ), demonstrating China’s reform and opening up and pioneering change in higher education.  要求使用图形用户界面，界面美观、交互友好。在报告中附上程序截图、运行结果和详细的文字说明。（5分）  程序由3个主要类组成：  1. *`TranslatorDemo.java`*：主程序类，实现GUI界面和核心逻辑  2. *`BaiduTranslator.java`*：百度翻译API工具类  3. *`AIModelTranslator.java`*：大模型翻译工具类  TranslatorDemo 类  主要功能：  - 创建和管理图形界面  - 处理用户输入和按钮事件  - 协调不同翻译方式的调用  关键代码：  \`\`\`java  public class TranslatorDemo extends JFrame {  private JTextArea chineseTextArea;  private JTextArea englishTextArea;  private JButton chToEngButton;  private JButton engToChButton;  private JComboBox<String> translationMethodCombo;    // 构造函数：初始化GUI界面  public TranslatorDemo() {  // ...初始化界面组件  }    // 翻译方法  private void translate(boolean isChToEng) {  // ...根据选择的方式进行翻译  }  }  **Part 3（30分）**  (3.1). 利用套接字连接（TCP）编写程序，该程序包括三个客户端（ClientA、ClientB、ClientC）和一个服务端（ServerS），三个客户端通过服务端作为桥梁实现相互间的文字交流，例如，ClientA先发信息给ServerS，然后ServerC再将收到的信息转发给ClientB和ClientC。在报告中附上示意图（三个客户端+一个服务端）、程序截图、完整的运行结果和简要文字说明。（20分）  (3.2). 利用数据报通信（UDP）实现题(1)中的要求。（10分）  ++++++++++++++++++++++++++++++++++++++++++++++++++++++  **其他（例如感想、建议等等）。** |

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。