四川大学计算机学院 （软件学院、 智能科学与技术学院）

实验报告

学号：2x 姓名： x 专业： 人工智能 班级： 233040701 第 9 周

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 课程名称 | 数据结构与算法分析课程设计 | | 实验课时 | 4课时 |
| 实验题目 | 简单文本编辑器 | | 实验时间 | 2024/10/24 |
| 实验目的和要求 | 基本要求  实现简单的文本编辑器，支持如下指令  R：加载文件  S: 写入文件  I：插入行  D：删除当前行  F：从当前行开始查找目标字符串  C：替换字符串  Q：退出编辑器  H：显示帮助  N：下一行  P：上一行  B：开始  E：结束  G：转到指定行  V：查看全部内容  已完成的附加要求：   1. 若有未保存的内容，退出时将提示 2. 支持撤销/重做操作   按下U将撤销，按下Z即可重做。 | | | |
| 实验环境 | Clion2024、MinGW11.0、C++17 | | | |
| 算法描述 | 本次实验使用了书上提到的CharString与LinkList类来简化程序的开发。  本程序的结构是main.cpp控制文件的打开，因为一个Editor类实例只会绑定一个源文件。  Main.cpp中有一个循环，其里面有一个try-catch块防止程序崩溃，这个循环开始时会提示用户输入要打开的文件信息，然后将这个文件名传入Editor类的构造函数，然后调用Editor类的beginEventLoop函数接管用户输入处理，这个函数有一个bool返回值，当当前Editor需要退出的时候的下一步操作，代表了需要打开新的文件还是退出整个程序。  对于Editor类，这个类是最复杂的包含了绝大部分文本操作。这个类包含了一个链表，文本文件中的每一行都将会分开存入这个链表中。其他函数部分如下：   1. **初始化和文件读取**：    * 构造函数Editor(const string& path)接受一个文件路径参数，并读取该文件内容，将每一行作为CharString对象插入到LinkList<CharString>类型的list中。如果文件为空，则插入一个空行。初始化后，会调用pushHistory将初始状态保存到历史记录中。 2. **历史记录管理**：    * pushHistory：保存当前编辑状态到历史记录中，如果历史记录中有historySnapshot指针后面发生的更改，则会先清除这些更改。    * undo：撤销上一次操作，通过将historySnapshot指针回退一个位置，并更新list到该历史状态。如果指针无法后退，就保持指针在开头。    * unundo（重做）：与undo相反，通过将historySnapshot指针前进一个位置，并更新list到该历史状态。如果指针无法前进，那么就保持指针在末尾。 3. **文件保存**：    * saveFile：将当前list中的内容保存到文件中，每行之间用换行符分隔。保存后将isUnsavedModifies标记为false，表示没有未保存的更改。 4. **文本查找**：    * findStr：从当前行开始向下查找指定的字符串，如果找到则打印位置并更新currentLine到找到的行。 5. **文本编辑**：    * makeChanges：在进行任何修改之前调用，用于保存当前状态到历史记录，并将isUnsavedModifies标记为true。    * change：替换当前行的内容，并标记为有未保存的更改。    * delete：删除当前行，并标记为有未保存的更改。    * insert：在指定位置插入一行，并标记为有未保存的更改。 6. **事件循环和命令处理**：    * beginEventLoop：是编辑器的主事件循环，它不断地读取用户输入的命令并执行相应的操作。支持的命令包括查看所有行、帮助、上下移动、文件开始和结束、跳转到指定行、保存文件、更改当前行、删除当前行、查找字符串、插入行、读取另一个文件、撤销和重做。 7. **辅助功能**：    * clearConsole：清空控制台。    * printHelp：打印帮助信息。 8. **异常处理**：    * 在处理用户输入时，如果遇到空命令或者未知命令，会抛出异常，并提示用户输入错误。   这个编辑器通过链表来管理文本行，支持基本的文本编辑功能，如撤销/重做、查找、插入、删除和保存。它还提供了一个简单的命令行界面，用户可以通过输入特定的命令来控制编辑器的行为。 | | | |
| 源程序清单 | main.cpp 程序入口与基本命令行交互  node.h 链表的节点  lk\_list.h 实现链表  char\_string.h 通过链表实现的字符串  Utils.h/Utils.cpp 定义工具类，实现文本文件读写，处理命令行输入等  Editor.h 定义文本编辑主程序 | | | |
| 运行结果 | 输入文件原式图下：    本程序运行如下：    实现的扩展要求如下：    更改后保存的文件如下： | | | |
| 实验运行情况分析(包括算法、运行结果、运行环境等问题的讨论) | **实验结果：**  实验最终运行结果与预期相符。  **问题的讨论：**  在设计撤销/重做功能时，最开始想的是使用一个标准的栈来实现，每次进行了内容更改操作就将当前的内容压栈，但是写着写着就发现标准的栈只能实现撤销功能，因为撤销功能与出栈操作相对应。为了能实现重做功能，我仔细思考了这个数据结构，于是我是用**vector容器来实现了一个魔改的栈**，使用了一个数字变量pointer来指向当前对应的状态，如果要撤销，只需要将pointer自减一，在这个时候，如果需要重做，将pointer自增即可（当然要检查边界状态）。如果在某刻发生了内容更改，需要先将pointer以后的状态全部删除（因为现在的状态被更改了，之前保存的后面的状态无效了，也就是说如果更改了当前状态，就不能重做了）。 | | | |
| 指导老师  评议 |  | | | |
| 成绩评定： | 指导教师签名： | | |