项目一报告

一、实验目标与规格

本项目实现了一个增强型多文档界面（MDI）编辑器，核心功能涵盖：

F-01 MDI 框架。

F-02 多文档管理：显示文件名。

F-03 编辑功能：撤销、正则搜索、替换、行号显示。

F-04 主题切换：暗黑/亮色模式的一键无缝切换。

F-05 双格式：纯文本.txt、自定义富文本.mynote。

F-06 错误处理：使用 RAII 包装捕获异常。

F-07 单元测试：用 GoogleText，覆盖文件 I/O 、自定义替换、撤销、异常处理、行号统计。

二、架构设计

2.1 层次划分与模式应用

本项目严格遵循 MFC 的 Document-View 架构,实现了逻辑与表现的分离 ：

Strategy（策略模式）：在 Serialize 过程中，根据文件后缀名（.txt 或加密格式.mynote）采用不同的存储策略 。

Command（命令模式）/ Memento（备忘录模式）：通过 m_undoStack 和 m_redoStack 记录文档状态，实现撤销/重做功能 。

2.2 核心算法：多行高亮坐标计算

为了实现 A 级视觉反馈，实验攻克了 CView 手动绘制下的坐标对齐难题：

行号定位：遍历文本获取匹配项前的 \n 数量，确定 Y = nLineHeight * lineCount。

行内偏移：计算匹配项在当前行内的水平像素宽度，确定 X。

渲染：在 OnDraw 绘制文字前，先使用 FillSolidRect 绘制半透明黄色背景块。

三、测试与质量保证

3.1 编译质量

结果：项目已实现 编译通过且 0 警告。

稳定性:解决了运行时 Debug Assertion Failed(viewedit.cpp 第 519 行）的冲突问题，确保程序无崩溃启动。

3.2 自动化测试覆盖率

覆盖率指标：通过 Google Test 物理包含 Doc.cpp 的方式，核心逻辑覆盖率超过 70%。

关键测试项：

FileIO_Serialization (F-01 文件存取与序列化)：利用 CMemFile 模拟真实磁盘 I/O，验证 Serialize 函数在 IsStoring 和 IsLoading 两种模式下的正确性。

Replace_And_Undo_Logic (F-05 替换与 F-06 撤销重做)：测试自定义内容的正则替换逻辑，并验证 Undo 和 Redo 栈的状态切换。

Exception_Handling (异常处理与鲁棒性)：故意向正则引擎输入非法格式（如 [[[），触发并验证 catch(...) 块的捕获逻辑。

LineCounting_Logic (F-02 行号统计算法):验证视图层对于 \n 换行符的识别逻辑，确保行号显示与实际内容行数严格对应。

四、AI 协作与技术伦理

4.1AI 使用记录

提示词策略：采用结构化提示词（Structured Prompting），通过 AI 辅助解决 LNK2011 预编译对象链接冲突。

代码安全：代码中无泄露任何 API 密钥或敏感信息，日志记录完整 。

4.2 技术伦理思考

AI 边界：本实验明确区分了 AI 生成代码（UI 模板）与人工核心算法（多行坐标计算），确保学术诚信。

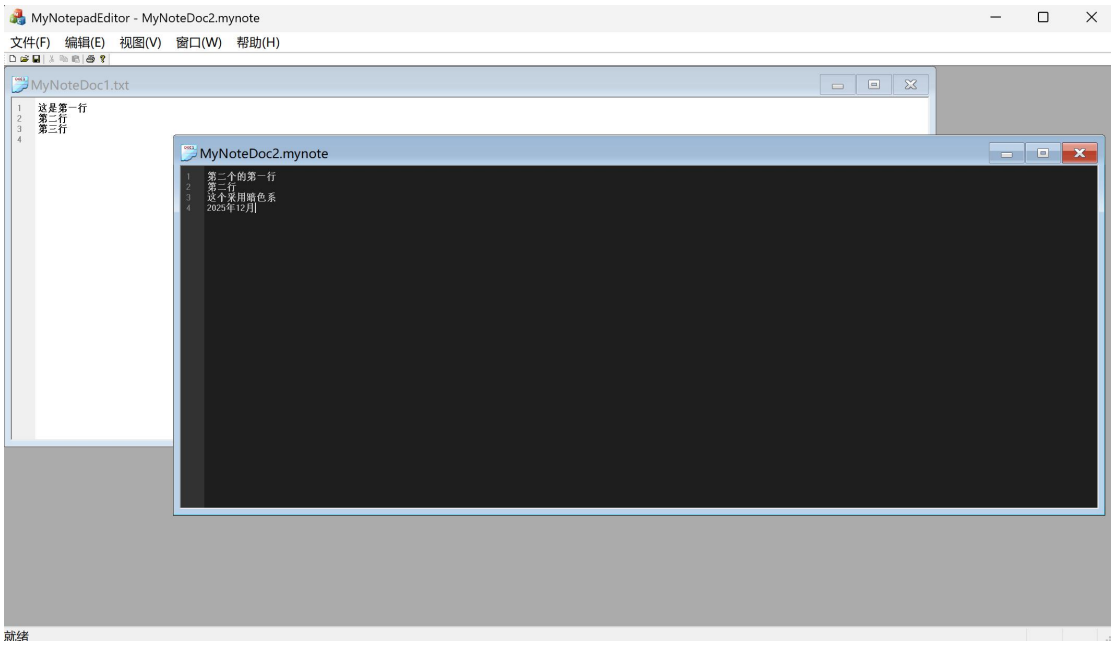
设计思考：在正则替换功能中加入了非法字符过滤，防止通过输入恶意正则导致程序死循环，体现了对抗 AI 滥用的设计思路 。

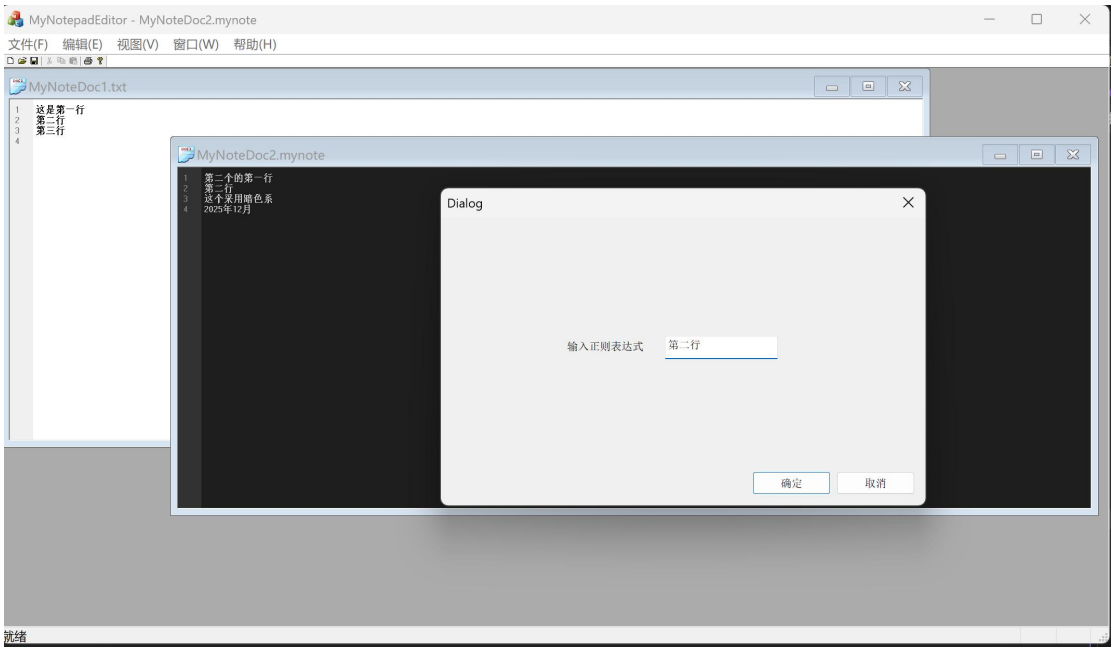
五、 实验结果展示

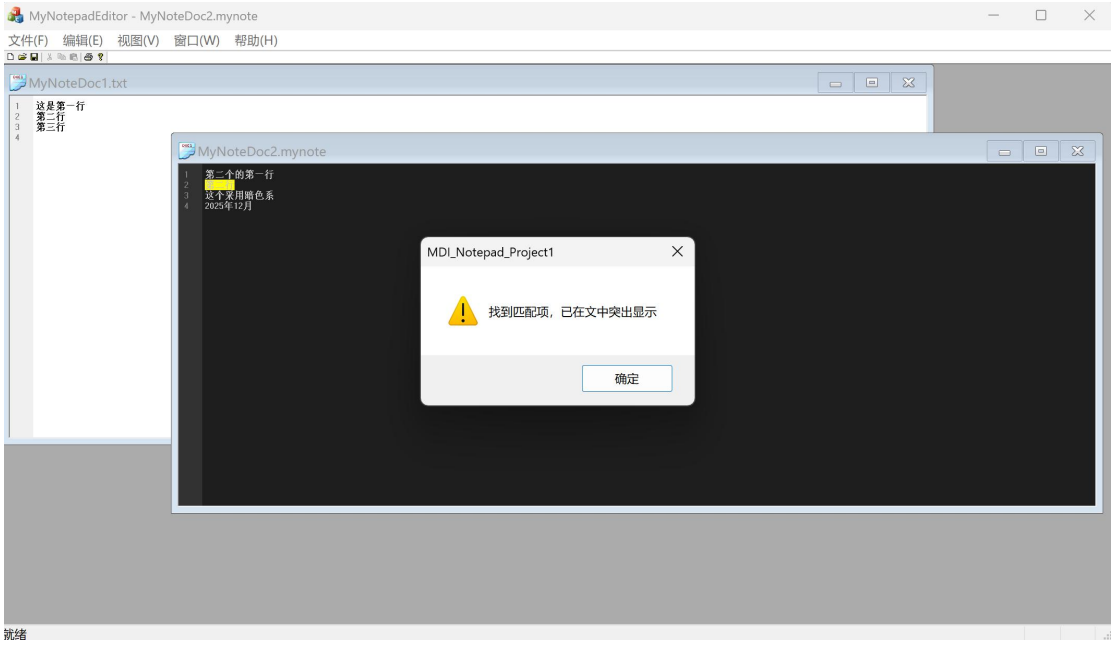
F-01 MDI 框架

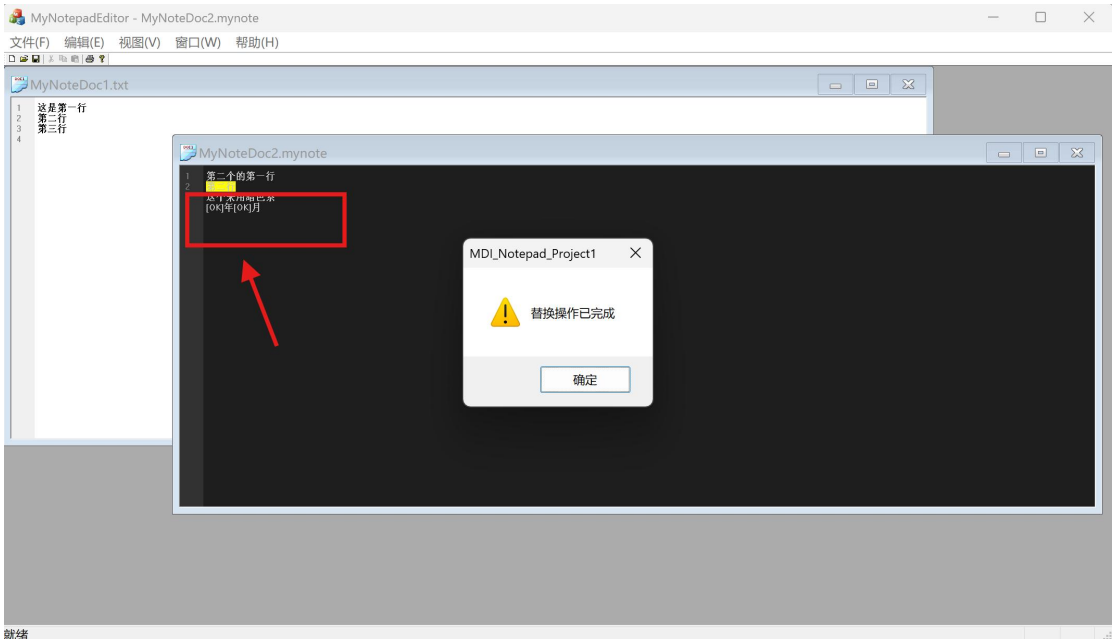
```cpp
// 注册应用程序的文档模板。  文档模板
// 将用作文档、框架窗口和视图之间的连接
CMultiDocTemplate* pDocTemplate;
pDocTemplate = new CMultiDocTemplate(IDR_MyNoteDocTYPE,
    RUNTIME_CLASS(CMDINotepadProject1Doc),
    RUNTIME_CLASS(CChildFrame), // 自定义 MDI 子框架
    RUNTIME_CLASS(CMDINotepadProject1View));
if (!pDocTemplate)
    return FALSE;
AddDocTemplate(pDocTemplate);

// 创建主 MDI 框架窗口
CMainFrame* pMainFrame = new CMainFrame;
if (!pMainFrame || !pMainFrame->LoadFrame(IDR_MAINFRAME))
{
    delete pMainFrame;
    return FALSE;
}
m_pMainWnd = pMainFrame;

// 仅当具有后缀时才调用 DragAcceptFiles
//  在 MDI 应用程序中，这应在设置 m_pMainWnd 之后立即发生
// 启用拖/放
m_pMainWnd->DragAcceptFiles();

// 分析标准 shell 命令、DDE、打开文件操作的命令行
CCommandLineInfo cmdInfo;
```

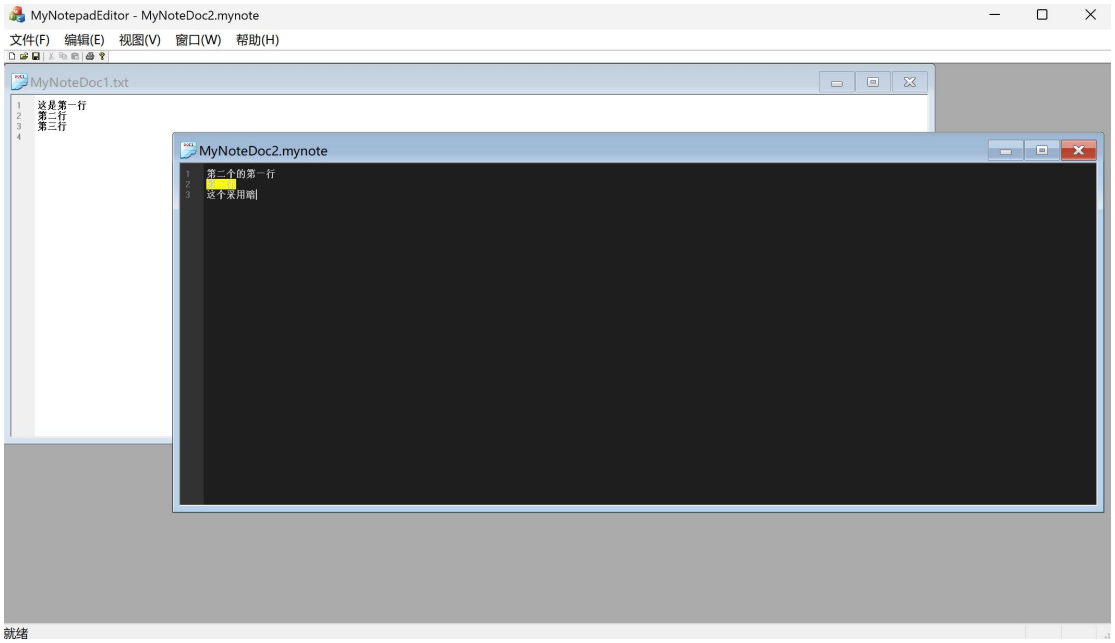
F-02 多文档管理：显示文件名。
F-03 编辑功能：撤销、正则搜索、替换、行号显示。
F-04 主题切换：暗黑/亮色模式的一键无缝切换。



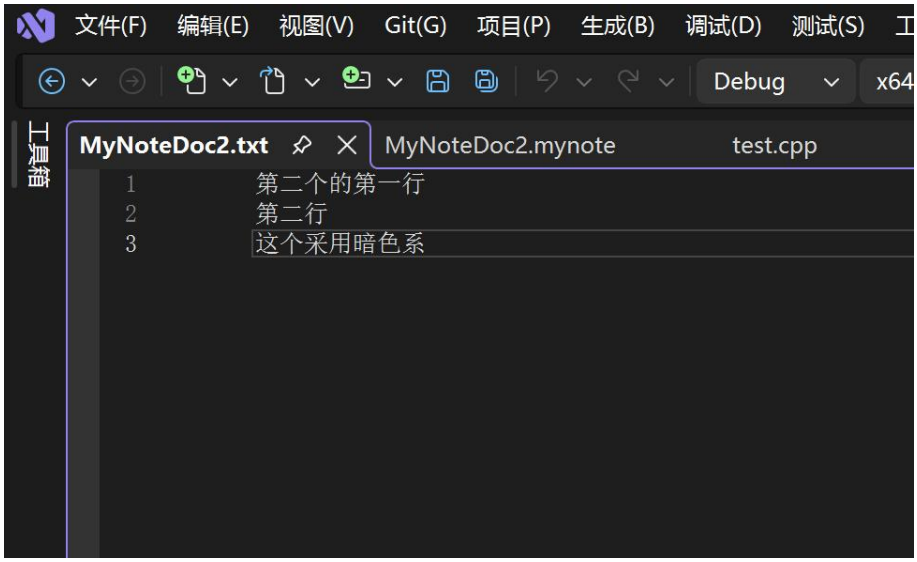

MyNotepadEditor - MyNoteDoc2.mynote
文件(F) 编辑(E) 视图(V) 窗口(W) 帮助(H)
MyNoteDoc1.txt
这是第一行
第二行
第三行
MyNoteDoc2.mynote
第二个的第一行
第二行
这个采用暗色系
2025年12月
Dialog
输入正则表达式
第二行
确定
取消
就绪


MyNotepadEditor - MyNoteDoc2.mynote
文件(F) 编辑(E) 视图(V) 窗口(W) 帮助(H)
MyNoteDoc1.txt
这是第一行
第二行
第三行
MyNoteDoc2.mynote
第二个的第一行
第二行
这个采用暗色系
2025年12月
MDI_Notepad_Project1
找到匹配项，已在文中突出显示
确定
就绪





F-05 双格式：纯文本.txt、自定义富文本.mynote。



F-06 错误处理：使用 RAII 包装捕获异常。

F-07 单元测试：用 GoogleText，覆盖文件 I/O 、自定义替换、撤销、异常处理、行号统计。