

湖南科技大学计算机科学与工程学院

软件工程导论

关于记事本应用的分析报告

学生姓名： 易成 陈建燊 易铭

学 号： 1705040207 1705040220 1705040217

专业班级： 17 级物联网工程二班

指导教师： 符琦

2019 年 11 月 11 日

目 录

第 1 章 记事本开源代码泛读报告.....	3
1.1 源代码文件架构说明.....	3
1.2 体系结构图.....	4
1.3 类间关系图.....	4
1.4 功能说明.....	5
1.5 应用功能与类间的对应关系.....	6
 第 2 章 记事本开源代码质量分析报告.....	 8
2.1 代码质量分析工具.....	8
2.2 分析结果.....	8
 第 3 章 记事本的维护需求与维护后效果演示.....	 10
3.1 维护需求描述.....	10
3.2 维护后效果演示.....	10

第 1 章 记事本开源代码泛读报告

1.1 源代码文件架构说明

记事本开源代码中类及各类的主要作用说明如表 1-1 所示。

文件名	子文件名	文件	主要作用
appimage	无	tkedit.desktop	设置应用图标、说明等
tkedit	languages	python.yaml	用 yaml 格式说明 python 中的关键字、变量等在编辑时可以按设定的文字颜色显示
		sql.yaml	用 yaml 格式说明 sql 中的关键字在编辑时可以按设定的颜色显示
	schemes	default.yaml	用 yaml 格式存储本应用默认的文字、背景颜色等
		font.yaml	用 yaml 格式存储在使用本应用时所编辑的文字字体及文字大小
	无	colourchooser.py	更换本应用外观、子菜单、编辑器等组件背景、前景颜色
	无	findwindow.py	寻找和替换目标字符
	无	fontchooser.py	更换字体种类及大小
	无	highlighter.py	使一个打开的文件中的相关字符高亮显示
	无	linenumbers.py	显示当前输入文字的行数
	无	requirements.txt	说明需要的格式是 yaml 文件格式
	无	setup.py	打包成 exe 文件的相关说明信息
	无	setup_cx_freeze.py	打包成 exe 文件所需的配置信息
	无	textarea.py	对应页面的编辑区以及 findwindow.py 中的查找、替换等功能的实现方法
	无	texteditor.py	创建主容器，包含一系列实现子菜单中

			定义的打开文件、存储所编辑的文件、 键盘快捷键等功能的方法
	无	tkedit.py	主方法入口，程序运行从这里开始
tkedit-snap	snap	snapcraft.yaml	用 yaml 格式存储本应用的说明信息
widgets	无	label_frame.py	主容器、子组件等的设定

表 1-1 源代码文件架构

1.2 体系结构图

本应用开发未采用标准 MVC 模式，其体系结构图如图 1.1 所示。

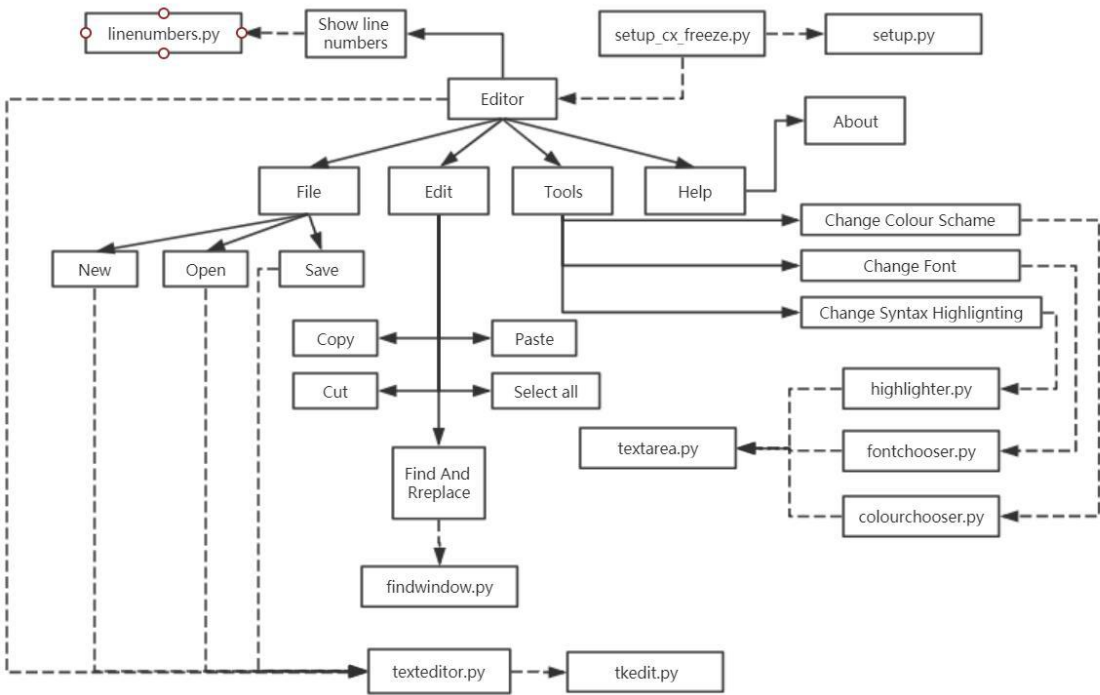


图 1.1 应用体系结构图

1.3 类间关系图

本应用业务处理部分和视图部分都在同一个文件夹 `schemes` 中，故类间关系图如图 1.2 所示。

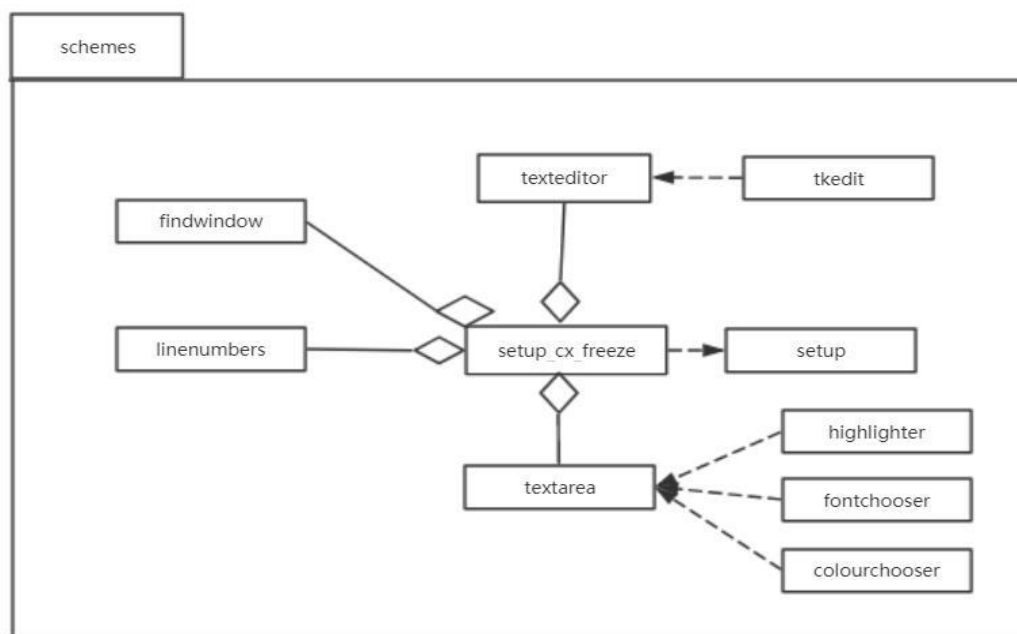


图 1.2 类间关系图

1.4 功能说明

本应用的功能架构图如图 1.3 所示。

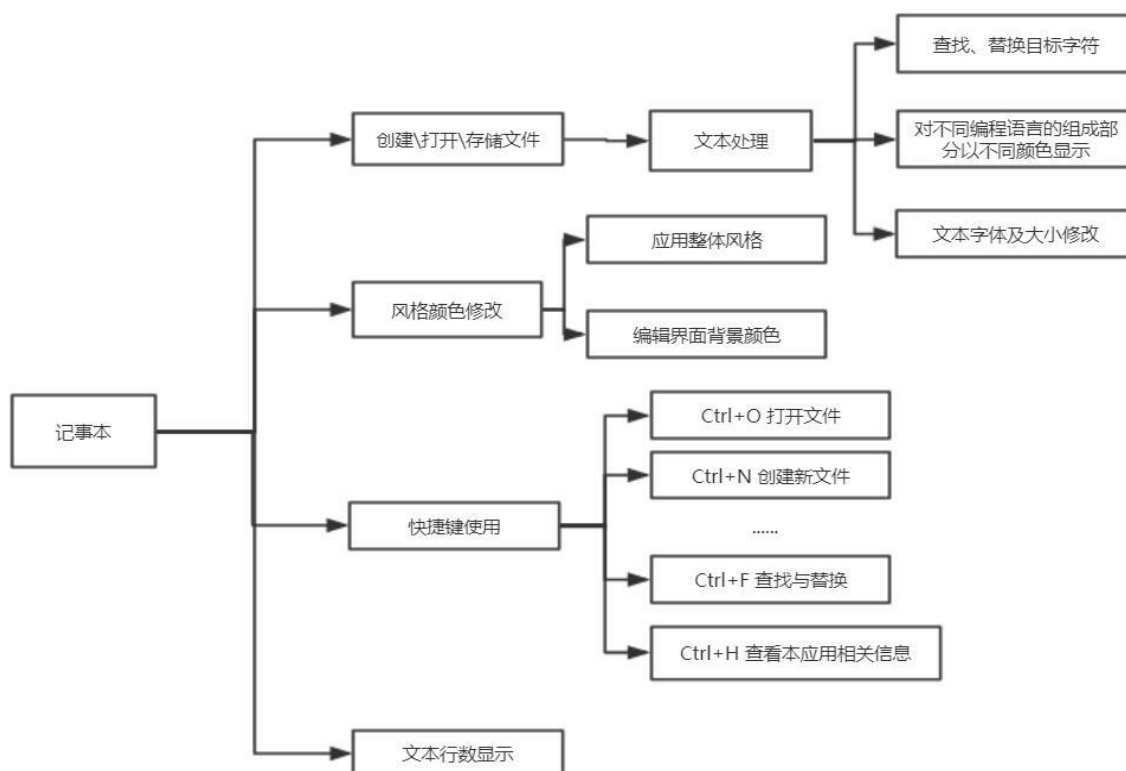


图 1.3 功能架构图

有关功能的详细说明如下：

(1) 创建\打开\存储文件：即用户可以通过点击相应功能按钮或按功能对应快捷键创建，打开，存储一个文件。

(2) 查找、替换目标字符：用户可以输入一个或一串字符，当点击“查找”按钮时，本应用会自动查找当前文本中是否存在用户输入的字符，如有，则提示用户查找成功，否则告知用户当前文本中不存在目标字符；当用户欲使用替换功能时，用户还需输入欲替换的字符，输入完成，点击“替换”即可完成目标字符的替换。

(3) 对不同编程语言的组成部分以不同颜色显示：本应用可以自动识别常规文字、数字、sql 和 python 语言中的不同组成部分（如关键字、变量等）并以不同颜色标明显示。

(4) 文本字体及大小修改：即对编辑区域所输入的文本的字体及字号大小进行修改。

(5) 应用整体风格\编辑界面背景颜色修改：即可以根据用户个人喜好修改应用整体风格以及编辑界面背景颜色。

(6) 快捷键使用：支持对创建文件，打开文件，存储文件，替换，存储等功能直接使用预先设定的快捷键。

(7) 实时显示文本行数：可以显示当前所输入的文本共有多少行。

1.5 应用功能与类间的对应关系

本应用中功能与类间的对应关系如表 1-2 所示。

序号	功能名称	实现模块	实现方法
1	创建\打开\存储文件	texteditor	file_new()、file_open()、 file_save()
2	查找、替换目标字符	textarea	find()、 replace_text()、 cancel_find()、 display_file_contents()
3	对不同编程语言的组成部分 以不同颜色显示	highlighter	highlighter 中所有方法

4	文本字体及大小修改	fontchooser	fontchooser 中的所有方法
5	应用整体风格\编辑界面背景 颜色修改	texteditor	load_scheme_file()、 change_colour_scheme()
6	快捷键使用	texteditor	bind_events()、edit_cut()等
7	实时显示文本行数	linenumbers	linenumbers 中的所有方法

表 1-2 功能与类间的对应关系

第 2 章 记事本开源代码质量分析报告

2.1 代码质量分析工具

本报告主要使用 SonarLint 工具对记事本应用软件进行质量分析，辅以少量的人工分析。

2.2 分析结果

本报告主要使用 SonarLint 工具对记事本应用软件进行质量分析，辅以少量的人工分析。通过对记事本所有源代码文件进行分析，得到如图 2.1 所示的代码质量分析结果。

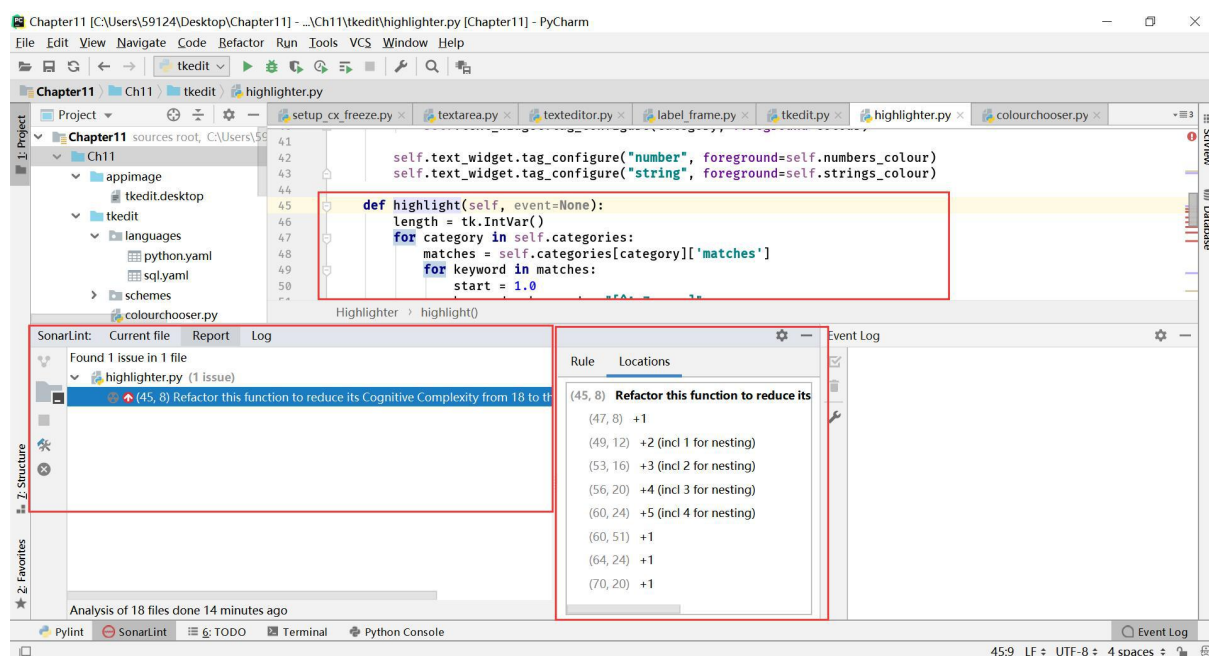


图 2.1 SonarLint 的分析结果

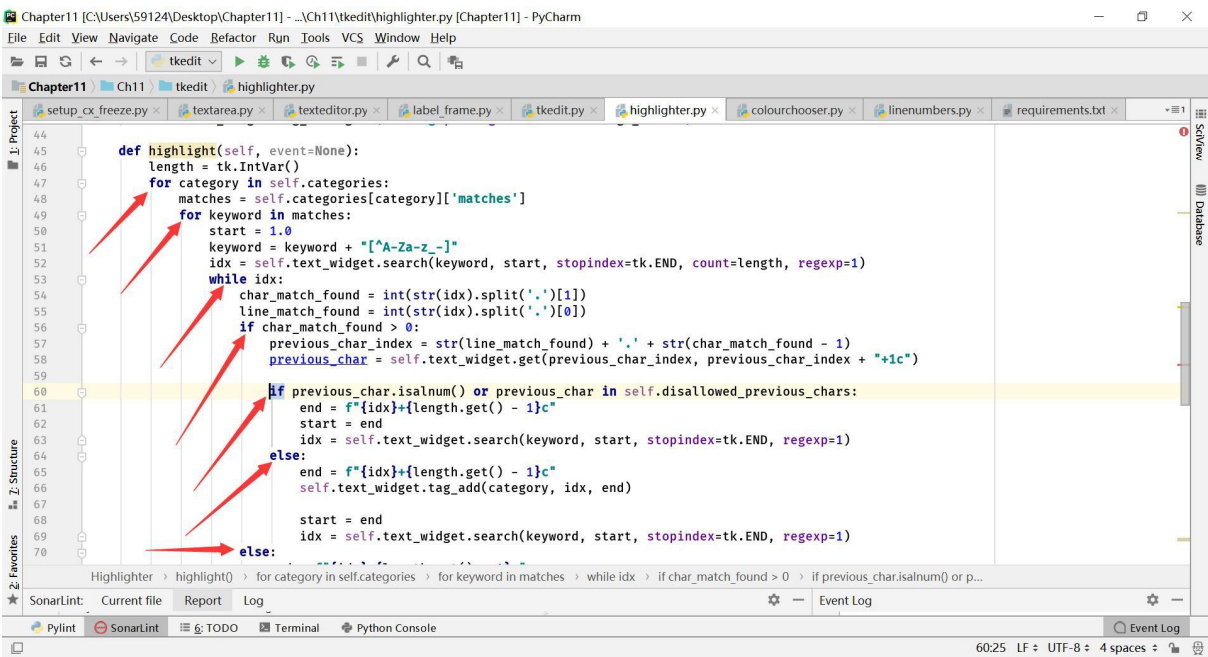
由于本应用软件规模不大，故在 SonarLint 审查的 18 个文件中只发现了 1 个 Critical（可能影响应用程序表现的错误和表示安全缺陷的问题）问题。经过人工分析，发现导致该问题的原因是 highlighter.py 文件下方法 highlight() 认知复杂度太高所导致的。

方法的认知复杂度是指衡量控制流理解难易程度的指标，可从以下方面进行评估：

代码块内部的嵌套控制流结构（对所有外层嵌套有影响）、使嵌套量增层数的控制流结构（for, while, do-while, if）和不使嵌套层数增加的控制流结构（else if, else, catch, 递归调用, 逻辑运算符序列）。如果多个语句在阅读时可以理解为一行，则忽略其对认知复杂度的影响（方法，是否为空的判断）。过高的认知复杂度可能会提高代码

维护的难度。

对于本软件，方法 `highlight()` 中的处理代码如图 2.2 所示。



```
44 def highlight(self, event=None):
45     length = tk.IntVar()
46     for category in self.categories:
47         matches = self.categories[category]['matches']
48         for keyword in matches:
49             start = 1.0
50             keyword = keyword + "[^A-Za-z_-]"
51             idx = self.text_widget.search(keyword, start, stopindex=tk.END, count=length, regexp=1)
52             while idx:
53                 char_match_found = int(str(idx).split('.')[1])
54                 line_match_found = int(str(idx).split('.')[0])
55                 if char_match_found > 0:
56                     previous_char_index = str(line_match_found) + '.' + str(char_match_found - 1)
57                     previous_char = self.text_widget.get(previous_char_index, previous_char_index + "+1c")
58                     if previous_char.isalnum() or previous_char in self.disallowed_previous_chars:
59                         end = f"{idx}+{length.get() - 1}c"
60                         start = end
61                         idx = self.text_widget.search(keyword, start, stopindex=tk.END, regexp=1)
62                     else:
63                         end = f"{idx}+{length.get() - 1}c"
64                         self.text_widget.tag_add(category, idx, end)
65                         start = end
66                         idx = self.text_widget.search(keyword, start, stopindex=tk.END, regexp=1)
67             else:
68                 start = end
69                 idx = self.text_widget.search(keyword, start, stopindex=tk.END, regexp=1)
70             else:
71                 start = end
72                 idx = self.text_widget.search(keyword, start, stopindex=tk.END, regexp=1)
```

图 2.2 `highlight()` 方法处理代码

即主要是因为处理代码中嵌套了 3 个循环，并且循环下还存在嵌套的分支语句，使得代码难以理解的程度增加（即认知复杂度增大）。

第3章 记事本的维护需求与维护后效果演示

3.1 维护需求描述

对当前版本记事本的功能分析发现，本软件仍存在以下几个问题：

- (1) 页面风格单一，不美观。
- (2) 功能“替换文本”不能正常使用。
- (3) 在“行数统计”功能中行数的显示存在滞后、错误显示、行号字体字号与文本不统一等现象。
- (4) 高亮显示异常，输入关键字要空格才能高亮，字符串中包含关键字部分会高亮而其他字符不亮，字符串高亮后，在字符串中加减字符后仍然高亮，问题很多。
- (5) 编辑区快捷键格式刷新异常。

另外，对于本应用，本组成员认为可以增加如下功能以提升软件实用性方便用户使用：

- (1) 增加计数功能，实时监测便签字数的变动，使用户可以随时了解当前便签的字数。
- (2) 增加字体风格和大小调整功能，可以选择自己喜欢的字体风格和合适的字体大小。
- (3) 增加撤销和恢复功能，可以回到上一步，也可以把撤销的恢复回来。
- (4) 增加一个终端，可以执行命令行的各行操作。

3.2 维护后效果演示

针对在“维护需求描述”部分中应用存在的功能问题，本组成员进行了改进与维护，详细效果如下所示：

- (1) 针对“功能‘替换文本’不能正常使用”，经维护后效果如图 3.1 和图 3.2 所示。

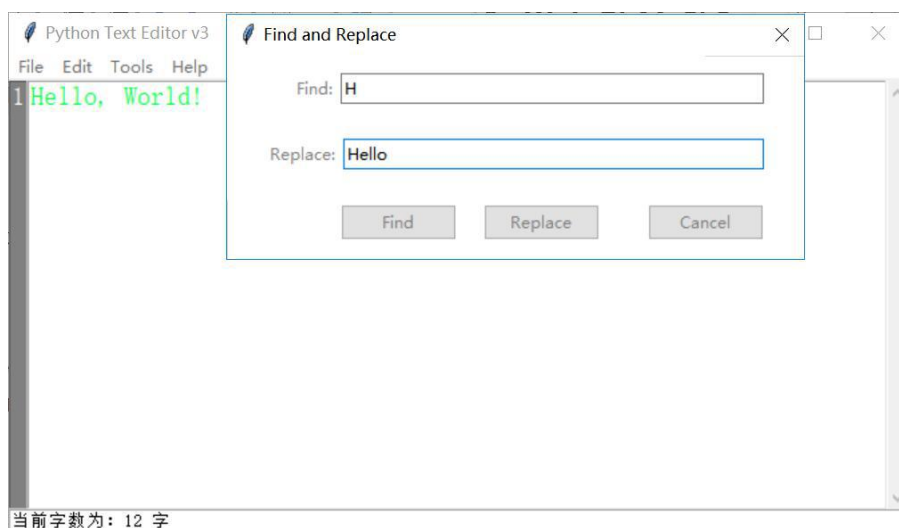


图 3.1 维护后效果

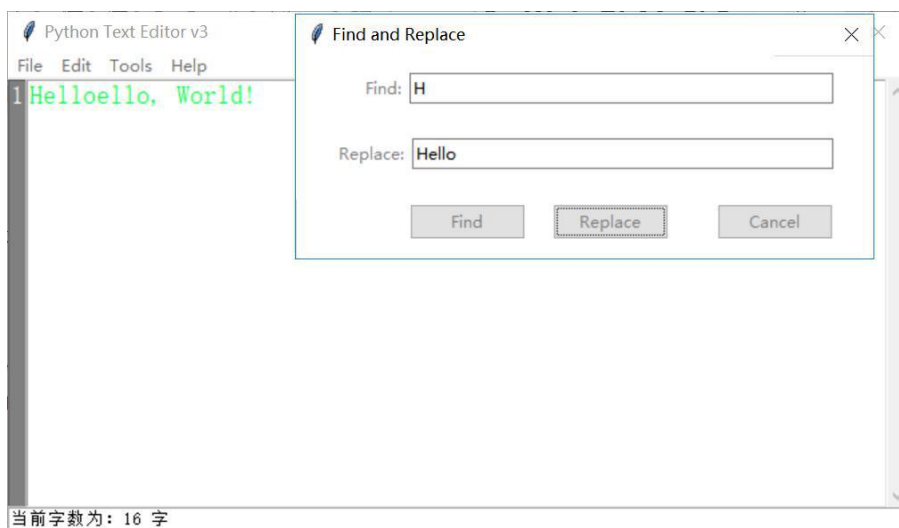


图 3.2 维护后效果

(2) 针对“‘行数统计’功能中行数显示问题”，经维护后效果如图 3.3 所示。

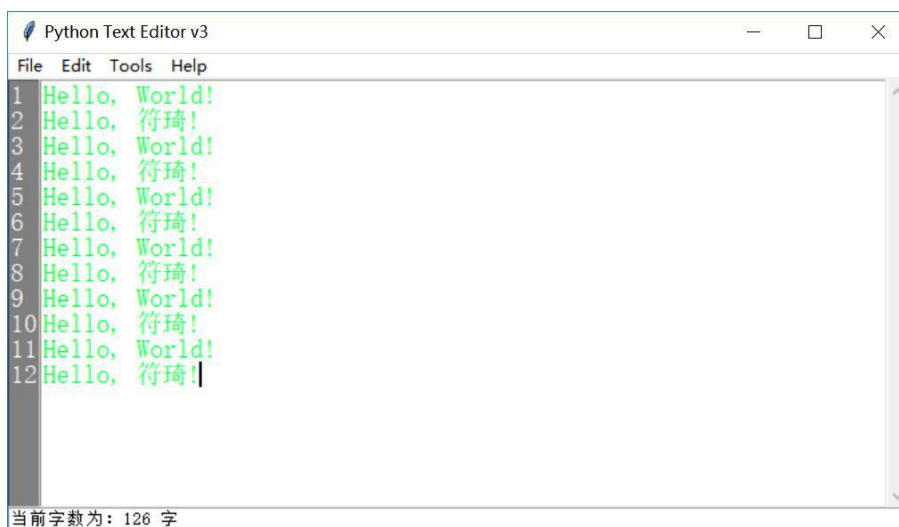


图 3.3 维护后效果

(3) 针对“增加计数功能”，经维护后效果如图 3.4 所示。

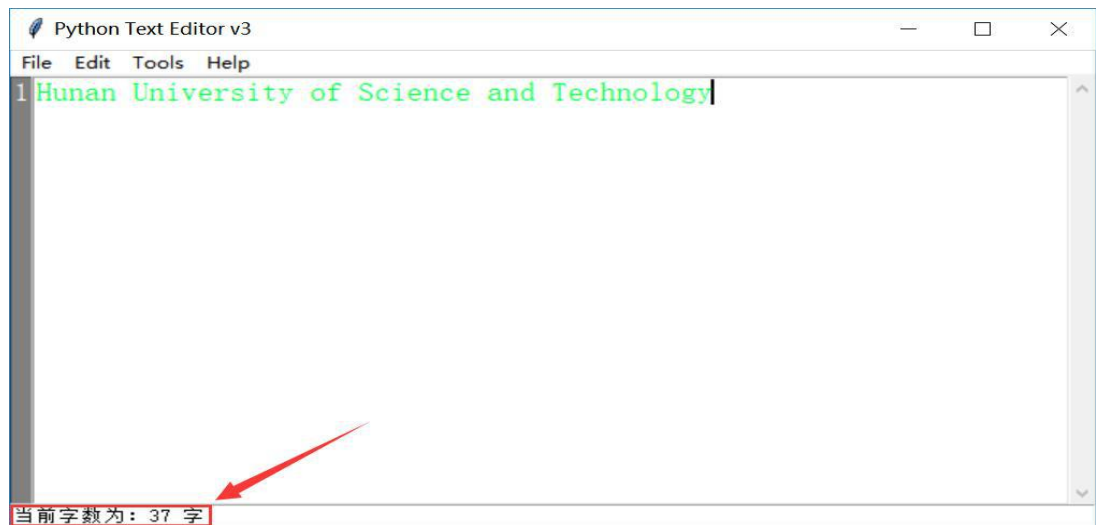
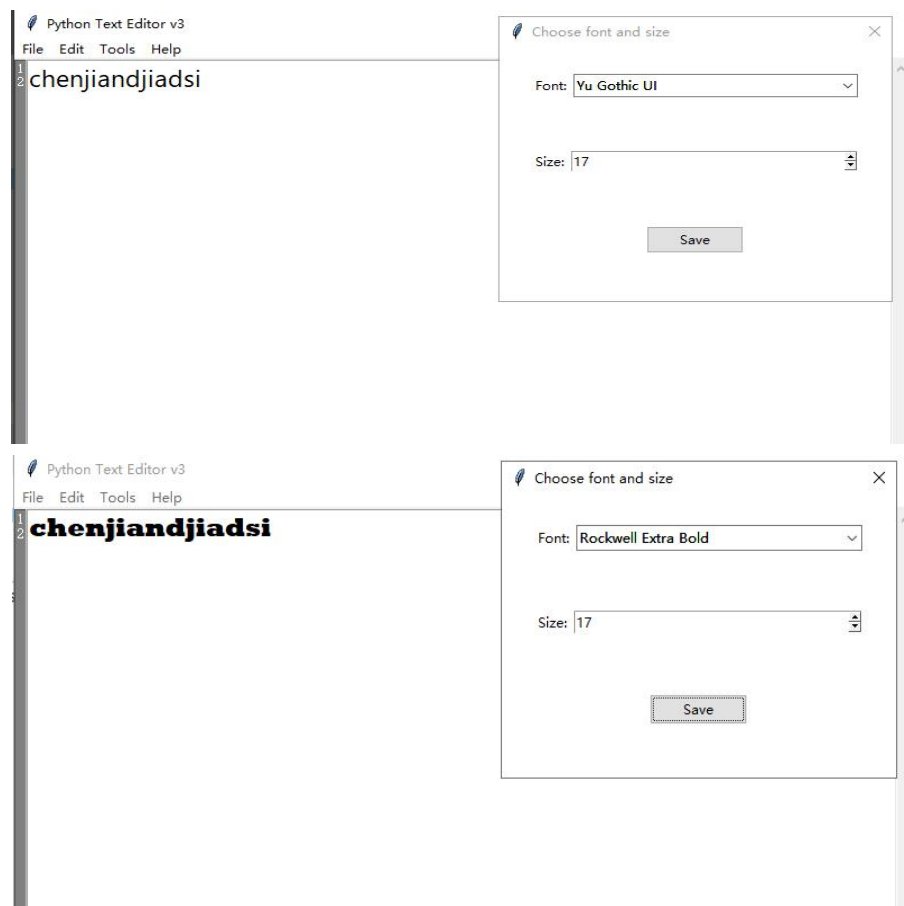


图 3.4 维护后效果

(4) 增加“修改字体和大小”，经维护后效果如图 3.5 所示。



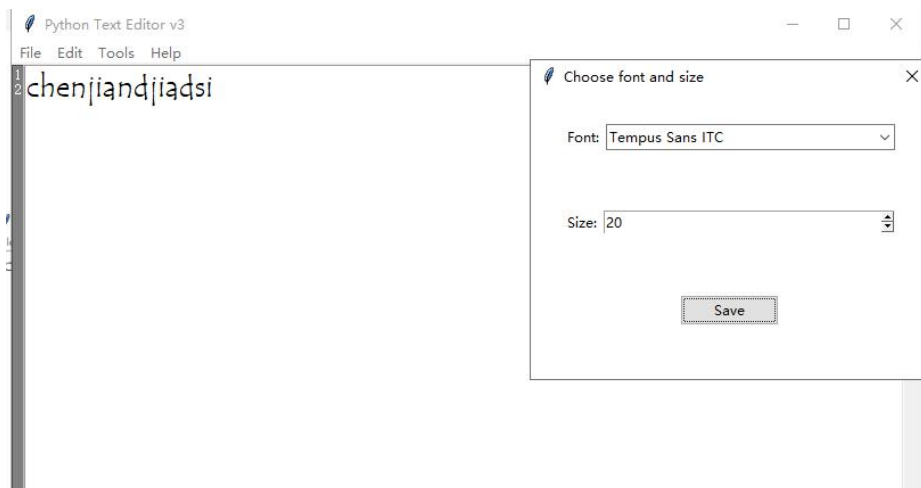


图 3.5 维护后效果

(5) 增加了撤销和恢复功能，效果如图 3.6 所示。

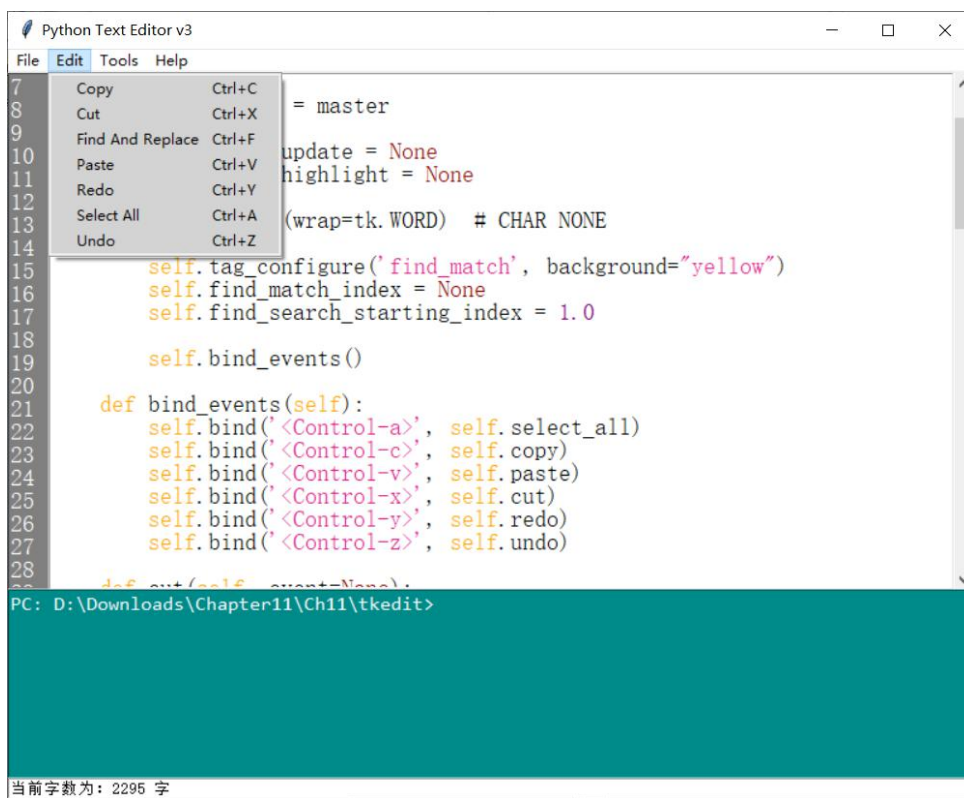


图 3.6 维护后效果

(6) 增加了一个终端，命令提示符和输出将不能被编辑，但可以被选择和复制。效果如图 3.7 所示。

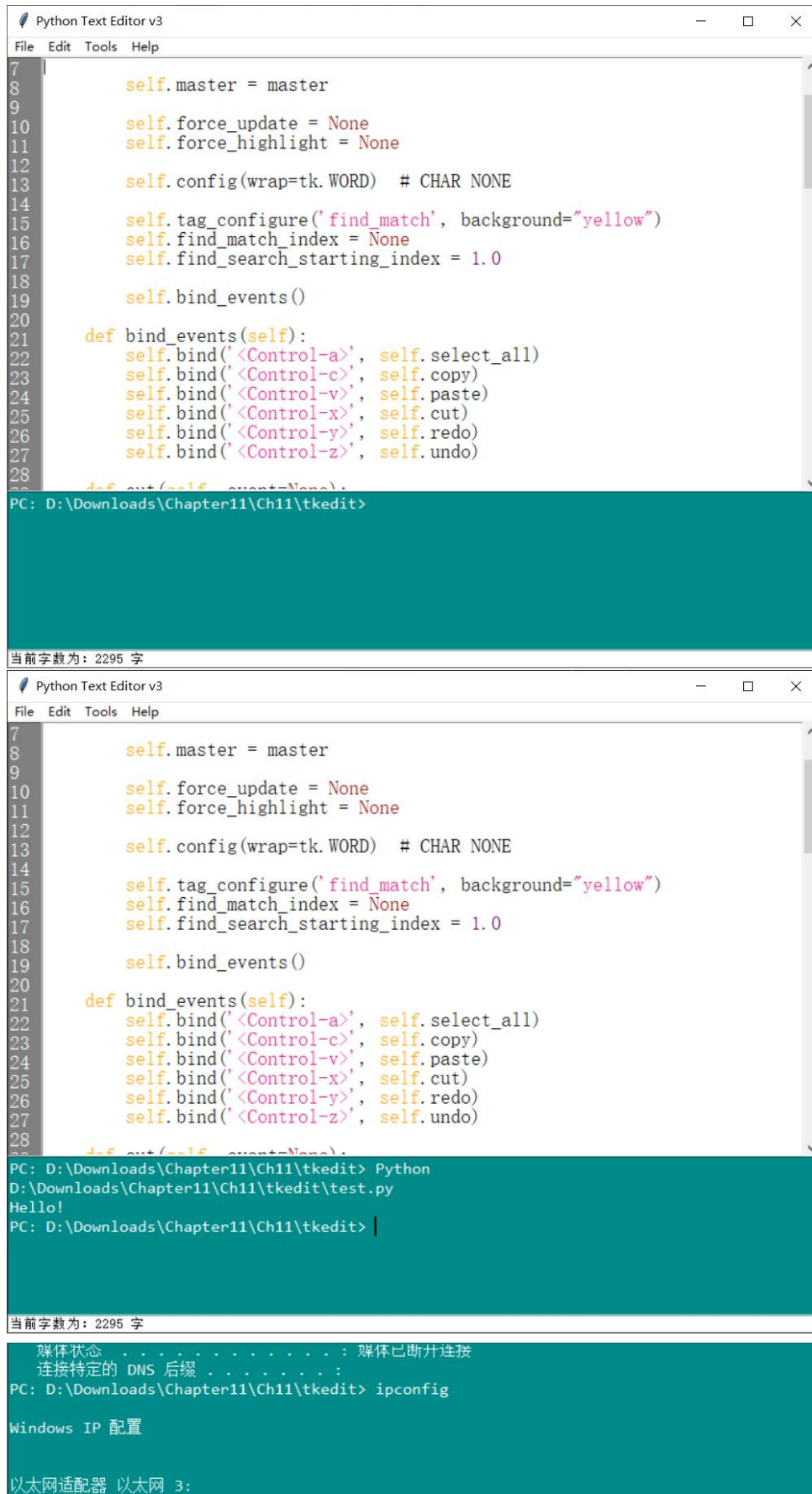
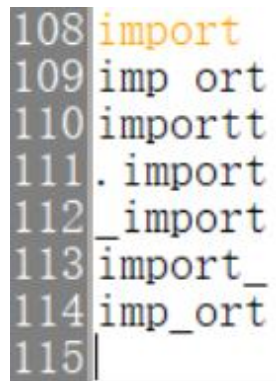


图 3.7 维护后效果

(7) 完全修复了行号和高亮异常，增加了 Up, Down, Page Up 和 Page Down 翻页，经维护后效果如图 3.8 所示。



```
108 import
109 imp ort
110 importt
111 .import
112 _import
113 import_
114 imp_ort
115 |
```

图 3.8 维护后效果

(8) 完全修复了编辑区快捷键格式刷新，详细效果见修复后的程序。