

CSCI 12042 Structured Programming II - Coursework

CS/2020/066

23/07/2023

—

Console Text Editor

—

GNU Utility URL

❖ GitHub

<https://github.com/YasinsaMahana/C-GNU-Utilities-Text-editor/tree/main>



Documentation and Naming

- ❖ The code includes comments explaining the purpose of various functions, variables, and sections of code, which is helpful for understanding the logic and flow of the program.
- ❖ Variable names 'MAX_F_L', 'MAX_T_L', 'f_name', and 'content' are adequately descriptive, making the code more readable.



Input Validation

- ❖ The code does not perform sufficient input validation. For example, the filename and content are directly taken from user input, which may lead to buffer overflow if the user enters strings longer than the defined maximum lengths (`MAX_F_L` and `MAX_T_L`).
- ❖ Additionally, the code should handle situations where the user enters invalid or incorrect filenames (e.g., with spaces or special characters). Consider adding input validation to prevent such issues.



Standard Input Function

- ❖ The code uses the 'scanf' function to read the filename, which may leave newline characters in the input buffer. It is a good practice to clear the input buffer after using 'scanf' to avoid potential issues with subsequent input operations.



User Interface

- ❖ The user interface is quite minimal and lacks proper formatting. Consider adding more descriptive and user-friendly messages for better interaction with the user.
- ❖ Provide a clear indication of when the user should input the text and the "exit" command.



Buffer Overflow

- ❖ The code uses a fixed-size character array to store the content of the file ('char content [MAX_T_L]'). There is no mechanism to handle cases where the content exceeds this fixed size, which can lead to buffer overflow issues. Consider using dynamic memory allocation or a more robust approach to handle variable-length content.



File Handling

- ❖ The code lacks error handling for file operations. It does not check if the file was successfully opened or closed. Always include proper error handling to handle scenarios where files cannot be opened or saved.



Platform-specific Functions

- ❖ The code includes the 'conio.h' header, which is not part of standard C and may not be available on all platforms. Avoid using platform-specific functions unless necessary for the specific platform.



Clearing Screen (Optional)

- ❖ Optionally, you can consider adding a function to clear the screen during different stages of the text editor's execution to provide a cleaner user interface.

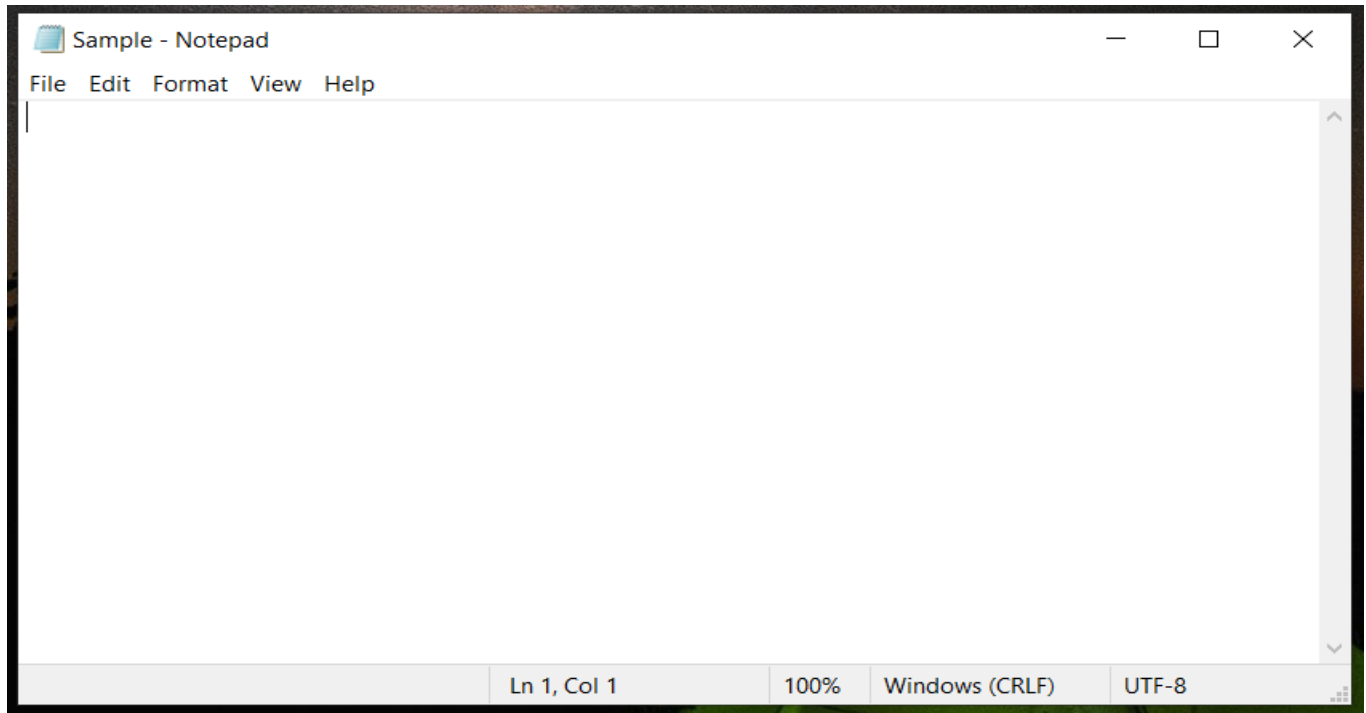


Feature Enhancement

- ❖ The current text editor lacks many standard features found in full-fledged text editors, such as editing existing files, handling multiple files, undo/redo functionality, find/replace, and syntax highlighting. Enhancements can be made to provide a more comprehensive text editing experience.



1.



2.



3.



```
"C:\Users\YASINSA MAHANAMA\Desktop\C-GNU\text_editor.exe"

.....CONSOLE TEXT EDITOR.....

Enter the filename(with extension & no spaces, ex: abc.txt) : Sample.txt

Enter the content
(type 'exit' on a new line to save and exit) :

.....
```

4.



```
"C:\Users\YASINSA MAHANAMA\Desktop\C-GNU\text_editor.exe"

.....CONSOLE TEXT EDITOR.....

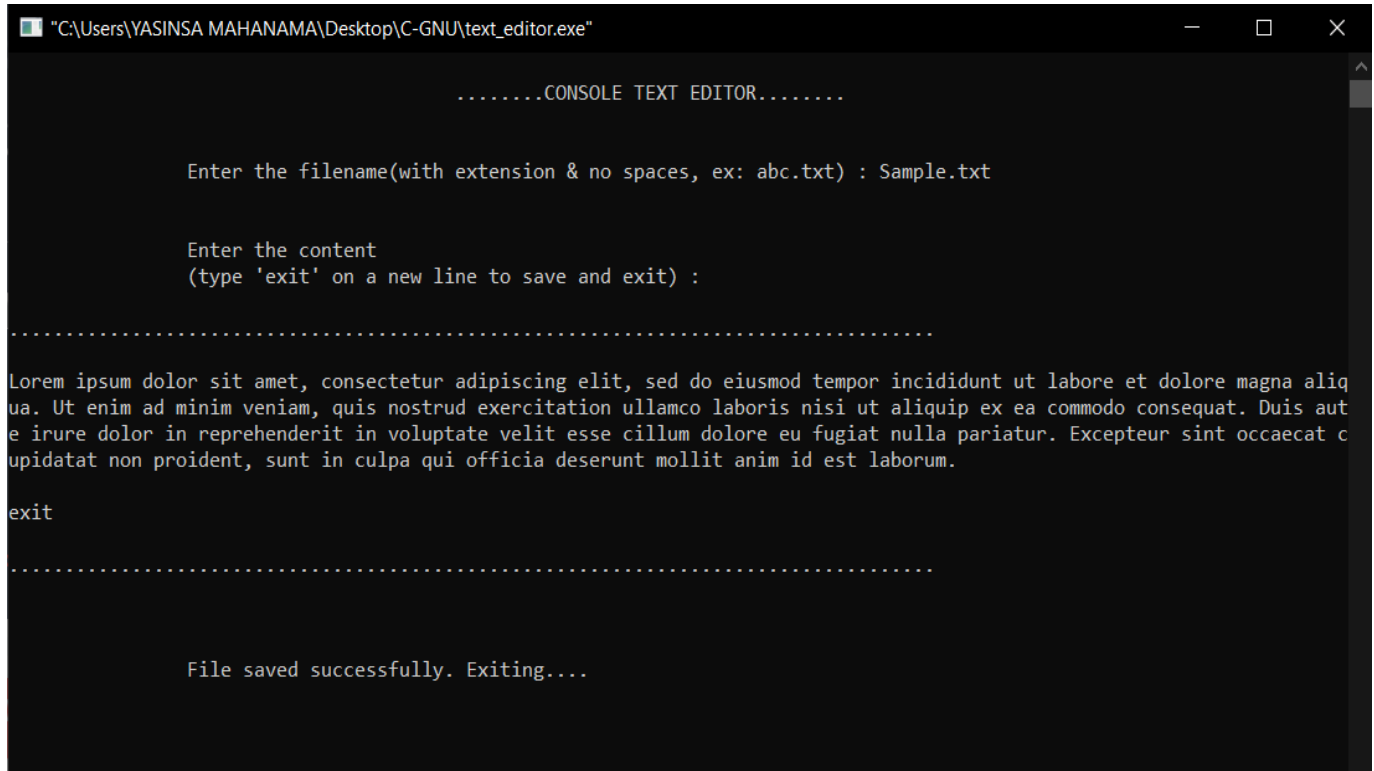
Enter the filename(with extension & no spaces, ex: abc.txt) : Sample.txt

Enter the content
(type 'exit' on a new line to save and exit) :

.....

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.
```

5.



```
.....CONSOLE TEXT EDITOR.....

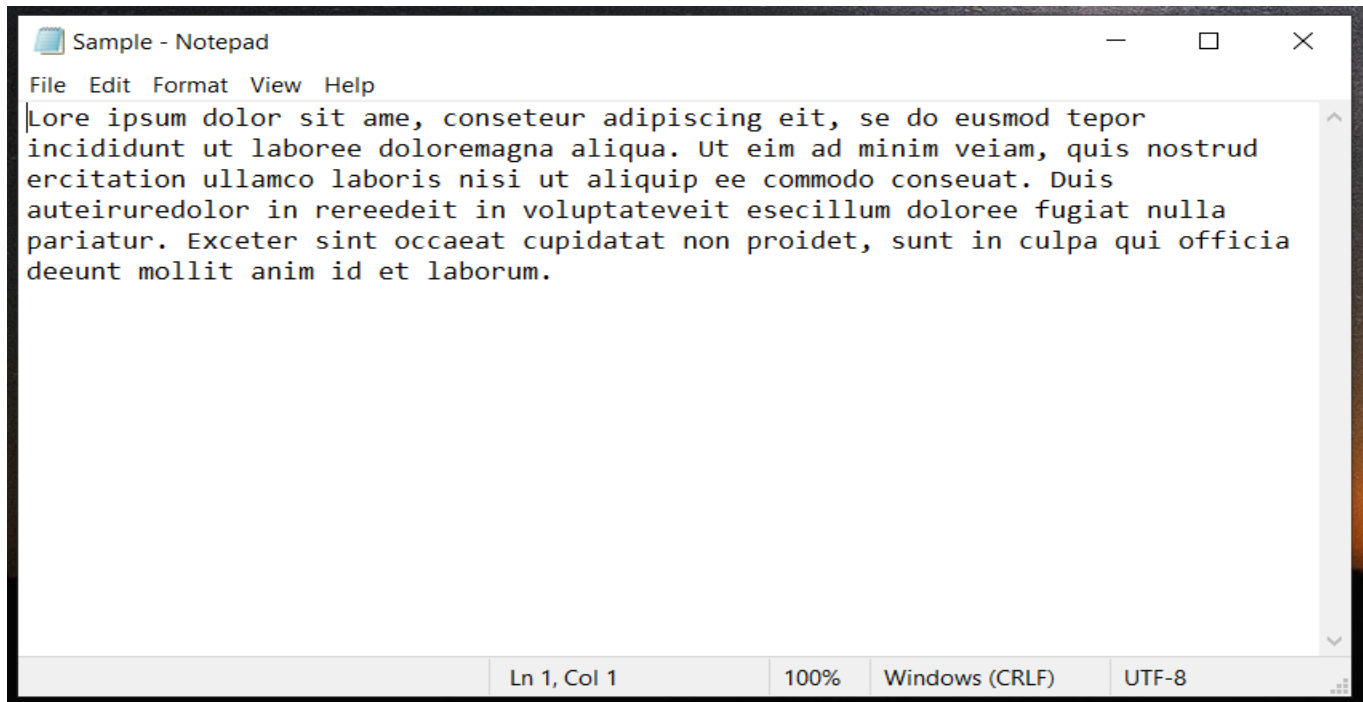
Enter the filename(with extension & no spaces, ex: abc.txt) : Sample.txt

Enter the content
(type 'exit' on a new line to save and exit) :

.....
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.
exit
.....

File saved successfully. Exiting....
```

6.



```
Sample - Notepad
File Edit Format View Help
Lorem ipsum dolor sit ame, conseteur adipiscing eit, se do eusmod tepor
incidunt ut laboree doloremagna aliqua. Ut eim ad minim veiam, quis nostrud
ercitation ullamco laboris nisi ut aliquip ee commodo conseuat. Duis
auteiruredolor in rereedeit in voluptateveit esecillum doloree fugiat nulla
pariatur. Exceter sint occaeat cupidatat non proidet, sunt in culpa qui officia
deeunt mollit anim id et laborum.

Ln 1, Col 1 100% Windows (CRLF) UTF-8
```


Overall, the provided code is a simple console-based text editor that allows users to enter text and save it to a file. However, it lacks many advanced features and robust error handling. Enhancements can be made to improve the code's functionality, user interface, and error handling for a more reliable and feature-rich text editor.

