**TERMINAL BASED TEXT EDITOR 2016**

Technical Manual

Team TermEdit

Github: mickmikea

Source code:

https://github.com/mickmikea/240-project.git

**Tables of Contents**

Program Overview 3

Program Structure 4

List of Source Files and Headers 5

List of Functions 6

Cross References 8

**Program Overview**

This program is a terminal based text editor created with the ncurses library. It has file saving and loading capabilities, and allows the user to scroll through large text documents, making changes as they please.

**Program Structure**

**List of Source Files and Headers**

**.CPP Files**

|  |  |
| --- | --- |
| main.cpp | Contains the main function, which is called at program startup |
| edit.cpp | Contains the functions of edit class |
| keybind.cpp | --- |
| read\_write\_file.cpp | Contains the functions of read\_write\_file class |
| undo.cpp | Contains the functions of undo class |

**.H Files**

|  |  |
| --- | --- |
| edit.h | Contains the definitions of the functions and attributes of edit class |
| keybind.h | Contains the definitions of the functions and attributes of keybind class |
| read\_write\_file.h | Contains the definitions of the functions and attributes of read\_write\_file class |
| undo.h | Contains the definitions of the functions and attributes of undo class |

**List of Functions**

**Main Functions (main.cpp)**

|  |  |
| --- | --- |
| main(int argc, char\*\* argv) | Creates the editor and executes the run method of the editor |

**Editor Functions (edit.cpp)**

|  |  |
| --- | --- |
| Editor::Editor(WINDOW\* window) | The editor class holds all of the main functions that make the text editor work |
| setupKeybindings() | set up the default key bindings for the editor |
| Editor::run() | Run the main editor loop that handles all key events for the editor |
| Editor::printLines() | prints all of the lines contained in the 'lines' vector to the console |
| Editor::drawStatusBar() | draw the status bar on the bottom of the terminal |
| Editor::checkLineBounds() | checks to be sure the cursor is within the bounds of the current line. If it's not, the cursorwill be moved to be at the end of the line |
| Editor::backspace(std::string& line, char keyPressed) | the callback for when backspace is pressed |
| Editor::newline(std::string& line, char keyPressed) |  |
| Editor::keyUp(std::string& line, char keyPressed) | Moves the cursor up one line, unless it is already at the top |
| Editor::keyDown(std::string& line, char keyPressed) | Moves the cursor down one line, unless the cursor is already at the bottom most line |
| Editor::keyLeft(std::string& line, char keyPressed) | Moves the cursor one space left |
| Editor::keyRight(std::string& line, char keyPressed) | Moves the cursor one space right |
| Editor::saveFile(std::string& line, char keyPressed) | Prompts the user to save a file |
| Editor::loadFile(std::string& fileName, char keyPressed) | Prompts the user to load a file |
| Editor::wordWrap() | Takes a spillover word and moves it down to the next line so that the line length remains at 80 characters |

**Functions to read and write a file (read\_write\_file.cpp)**

|  |  |
| --- | --- |
| read\_write\_file::read\_file(std::string name) | takes a file name and returns a vector of the file info |
| read\_write\_file::write\_file(std::vector<std::string> text, std::string name) | takes a vector and a filename and writes a vector to that filename |

**Cross Reference**

**Editor functions (edit.cpp)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function** | **Used in** | | **Functions called within** | |
| Editor::Editor(WINDOW\* window) | | -- | | keypad(window, true)  setupKeybindings() |
| Editor::setupKeybindings() | | Editor::Editor(WINDOW\* window) | | Editor::backspace()  Editor::newLine()  Editor::keyUp()  Editor::keyDown()  Editor::keyRight()  Editor::keyLeft()  Editor::saveFile()  Editor::loadFile() |
| Editor::run() | | main.cpp | | Editor::wordWrap()  Editor::printLines()  Editor::drawStatusBar() |
| Editor::printLines() | | Editor::run()  Editor::saveFile()  Editor::loadFile | | -- |
| Editor::drawStatusBar() | | Editor::run()  Editor::saveFile()  Editor::loadFile | | -- |
| Editor::checkLineBounds() | | Editor::keyUp()  Editor::keyDown() | | -- |
| Editor::backspace(std::string& line, char keyPressed) | | Editor::setupKeybindings() | | -- |
| Editor::newLine(std::string& line, char keyPressed) | | Editor::saveFile()  Editor::loadFile()  Editor::setupKeybindings() | | -- |
| Editor::keyUp(std::string& line, char keyPressed) | | Editor::setupKeybindings()  Editor::checkLineBounds() | | -- |
| Editor::keyDown(std::string& line, char keyPressed) | | Editor::setupKeybindings()  Editor::checkLineBounds() | | -- |
| Editor::keyLeft(std::string& line, char keyPressed) | | Editor::setupKeybindings() | | Editor::keyUp(line, keyPressed) |
| Editor::keyRight(std::string& line, char keyPressed) | | Editor::setupKeybindings() | | Editor::keyDown(line, keyPressed) |
| Editor::saveFile(std::string& line, char keyPressed) | | Editor::setupKeybindings() | | Editor::printLines()  Editor::drawStatusBar() |
| Editor::loadFile(std::string& fileName, char keyPressed) | | Editor::setupKeybindings() | | Editor::printLines()  Editor::drawStatusBar()  Editor::wordWrap() |
| findLastSpace(std::string str) | | Editor::wordWrap() | | -- |
| Editor::wordWrap() | | Editor::loadFile()  Editor::wordWrap() | | findLastSpace() |

**Functions to read and write files (read\_write\_files.cpp)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function** | **Used in** | | **Functions called within** | |
| read\_write\_file::read\_file(std::string name) | | Editor::loadFile() | | -- |
| read\_write\_file::write\_file(std::vector<std::string> text, std::string name) | | Editor::saveFile() | | -- |