

# Building Your Own Vim-like Text Editor

Contributor: 张泽凯 [12442026@mail.sustech.edu.cn](mailto:12442026@mail.sustech.edu.cn)

**Welcome, future Vim-masters!** Today, you begin your journey toward creating a mini version of Vim, the mighty text editor loved by programmers across the globe. Originally birthed in the '70s as **Vi**, and then leveled up in 1991 as **Vim** ("Vi IMproved"), Vim lets users dive deep into their code with pure keyboard power. Though it has a reputation for being challenging to learn, many find that mastering Vi/Vim significantly boosts productivity and control when working with text.



## Project Overview

The project is divided into two levels. The basic level satisfies the essential requirements for functionality, while the advanced level is for those who want to push the boundaries and implement additional features.

## Basic Requirements

Vim has three primary modes: **Normal**, **Insert**, and **Command-Line**. Each mode serves a specific purpose.

1. **Normal Mode** - Navigate, edit, and more.
2. **Insert Mode** - Type and modify text.
3. **Command-Line Mode** - Save, search, or quit with commands like `:w`, `:q`, and more.

## Normal Mode

Normal Mode is the **default** mode when you open Vim. In Normal mode, you're not directly typing text but instead issuing commands to navigate, delete, copy, paste, and manipulate text. This mode allows you to quickly perform complex text operations with minimal keystrokes. Think of it as the "control center".

- **File and Mode Commands:**
  - `:w` - Save the file.
  - `:q` - Quit the editor.
  - `:wq` - Save and quit.
  - `i` - Switch to Insert mode.
  - Press `:` to open **Command-Line mode** for more advanced commands.
- **Cursor Movement:**
  - Move with `h` (left), `j` (down), `k` (up), and `l` (right).
  - Arrow keys also work for cursor navigation.
- **Line Navigation:**
  - `0` - Jump to the beginning of the line.
  - `$` - Jump to the end of the line.
  - `gg` - Go to the first line.
  - `G` - Go to the last line.
- **Basic Editing:**
  - `dd` - Delete the current line.
  - `yy` - Copy the line.
  - `p` - Paste the copied line below the cursor.

## Insert Mode

Insert Mode allows direct text input. Here, everything you type appears on the screen (like a traditional editor). As mentioned above, activate it by pressing `i` in Normal Mode, and return to Normal Mode using `Esc`.

- **Direct Text Insertion:** Type characters that display at the cursor position.
- **New Line Creation:** `Enter` inserts a new line, moving the cursor to the beginning of the new line.
- **Backspace Support:** Delete characters with `Backspace`. Navigate up to delete across lines if at the cursor is at the start of a line.
- **Real-Time Display Update:** Changes should instantly reflect in the text.
- **Cursor Movement with Typed Text:** Cursor should keep up with typing, mimicking typical editor behavior.
- **Exit to Normal Mode:** `Esc` should switch back to Normal mode.

## : Command-Line Mode

Command-Line mode is used for file operations, search, and more. To enter Command-Line mode from Normal mode, simply press `:`, then complete commands such as `:w` (save), `:q` (quit), or `:wq` (save and quit), and press `Enter` to execute them. Note that Command-Line mode has been implemented as part of the requirements in Normal mode.

# Advanced Features

Here are some ways to make your mini-Vim truly powerful! Below are some possible features you can work on, and you may **customize them as you wish**, as long as they make sense for the project.

- **Search and Replace:** Implement find and replace with a command like `:s/old/new/g`.
- **Multi-File Editing:** Open and switch between multiple files seamlessly.
- **Undo and Redo:** Let users undo or redo their last action.
- **Line Numbers and Jumping:** Show line numbers and jump to a specific line.
- **UI Improvements:** Add some pizzazz! Use colors or styles to make different modes or commands stand out.

## Submission Requirements

- **Source Code:** Submit all relevant source files for the project.
- **Project Documentation:** Write a **brief** doc that explains your design choices, features, and usage instructions.

## Grading Criteria

Category	Criteria	Points
Basic Requirements (80 points)		
Normal Mode & Command-Line Mode (40 points)	<b>File and Mode Commands</b>	<b>20</b>
	- <code>:w</code> Save file	4
	- <code>:q</code> Quit editor	4
	- <code>:wq</code> Save and quit	4
	- <code>i</code> Switch to Insert mode	4
	- <code>:</code> Open Command-Line mode	4
	<b>Cursor Movement</b>	<b>10</b>
	- Move with <code>h/j/k/l</code>	4
	- Arrow key navigation	2
	- <code>0</code> and <code>\$</code> for line start and end	2
	- <code>gg</code> and <code>G</code> for first and last line navigation	2
	<b>Line Operations</b>	<b>10</b>
	- <code>dd</code> Delete current line	4
	- <code>yy</code> Copy current line and <code>p</code> Paste copied line	6
Insert Mode (25 points)	<b>Text Insertion</b>	<b>20</b>
	- Type characters at cursor position	5

	- Enter capabilities	5
	- Backspace capabilities	5
	- Real-time cursor updates during typing	5
	<b>Mode Switching</b>	<b>5</b>
	- Press <code>Esc</code> to return to Normal mode	5
<b>Project Completeness and Documentation (15 points)</b>	<b>Source Code Completeness</b>	<b>5</b>
	- All files compile and run correctly	5
	<b>Documentation</b>	<b>10</b>
	- Brief and clear design and usage explanation	10
<b>Advanced Features (Maximum 40 points)</b>	<b>Search and Replace</b>	<b>10.</b>
	- Implement <code>:s/old/new/g</code> command	10 Max.
	<b>Multi-File Editing</b>	<b>20</b>
	- Support for opening and switching between files	20 Max.
	<b>Undo and Redo</b>	<b>10</b>
	- Implement <code>u</code> (undo) and <code>Ctrl+R</code> (redo)	10 Max.
	<b>Line Numbers and Jumping</b>	<b>10</b>
	- Show line numbers and jump to specific lines	10 Max.
	...	

## Hint

You might consider using `<ncurses.h>` for terminal-based capabilities. Here are some good resources to start with:

- NCURSES Programming HOWTO: <https://tldp.org/HOWTO/NCURSES-Programming-HOWTO/>
- NCURSES Programming HOWTO 中文翻译: <https://poet.cosoft.org.cn/downloads/docs/ncurses-cn-2nd.pdf>

If you are not familiar with Vi/Vim, try

- an interactive tutorial: <https://openvim.com/>

---

Good luck! After this project, you might be the one laughing at all those "How to Exit Vim" memes. 😊