MODULE - IV

Vi Editor

Vi Editor

Module Description

The vi (visual editor) is the first full screen editor of UNIX. It allows user to view and edit the entire document at the same time. The main goal of studying vi editor is to enable students to write and edit C programs or shell scripts efficiently in an editor. This module concentrates on text processing invi editor, modes in vi editor, process of yanking, shell command using macros and basic elements of communication process.

By the end of this module, students will learn the modes in which vi operates, search for a pattern and repeat the search, various window properties with vi editor, Identify the basic elements of communication process, Use of various commands to communicate with other users, Demonstrate how UNIX administrators communicate with other users using mail, wall, send, mesg, and ftp.

Chapter 4.1
Introduction to Vi Editor

Chapter 4.2
Advanced features of Vi Editor

Chapter Table of Contents

Chapter 4.1

Introduction to Vi Editor

Aim

To enable students to view, write and edit documents efficiently using vi editor

Instructional Objectives

After completing this chapter, you should be able to:

- Define the concept of text processing in vi editor

- Explain the different modes in vi editor

- Illustrate the procedure to switch from one mode to another mode

- Demonstrate how to delete and insert line in a file

- Demonstrate how to delete and replace character in a file

- Illustrate how to search and replace a string in vi editor

- Explain the process of yanking

Learning Outcomes

At the end of this chapter, you are expected to:

- Outline the features of vi editor in text processing

- Perform the commands used to save and quit a file

- Perform the commands insert a text, append a text and move cursor position

- Define the steps to change from insertion mode to execution mode

- Perform the commands used for deleting and replacing character in a file

- Demonstrate string related commands and its purposes

- Write the steps to yank a line of text and paste the copied contents to the last line

4.1.1 Introduction

Vi is a text editor on Unix. Bill Joy created this editor for BSD system. This editor is now standard on all UNIX systems. Like any other editor, vi allows you to move to any location in the file and do editing. It also allows you to copy the content within a file or between different files. To understand it, working knowledge of vi editor is required at initial level which is discussed in this chapter. The advanced features of vi are discussed in chapter 4.2.

4.1.2 Introduction to Text Processing

Text processing on any OS involves creating and saving a text file, navigating the file and edit text at the cursor position. It also involves copying, moving text from one place to another and deleting characters in a file. Vi provides some internals commands for editing work.

4.1.3 Command Mode and Edit Mode

Vi editor has two modes of operation

- Command Mode : It is used to perform following operations:
  - saving files,
  - executing commands,
  - navigate in the document i.e. moving cursor,
  - cutting, yanking (copying) and pasting lines or words,
  - finding or replacing.
  - In this mode whatever text typed is treated as command.
- Edit Mode or Insert Mode: It is used to do text editing in the document.
  - In this mode whatever text typed is treated as input text to save in the file.

The Vi always starts in command mode. To enter or edit text, you must be in the insert mode.

Tips
- Press i - to go into the insert mode.
- Press ESC – to come out of the insert mode i.e. to command mode.
- Important: If you are not sure in which mode you currently are, press ESC twice, you would be in the command mode.

4.1.4 Invoking Vi

Type vito start vi editor in the terminal window

- **Creating a new file in the Vi Editor**

Snapshot (1)

Figure 4.1.1

It creates a new file if does not exist or open an existing file

It open the vi with file name at the bottom of the screen with information tag [New File] as shown below. Rest of the lines show ~ sign at the beginning of each line. These are virtual lines which do not exist, so you cannot navigate to these lines.

Snapshot (2)

Figure 4.1.2

Now, you can edit text in the file by entering into the Edit or insert mode by pressing **i,** as shown below:
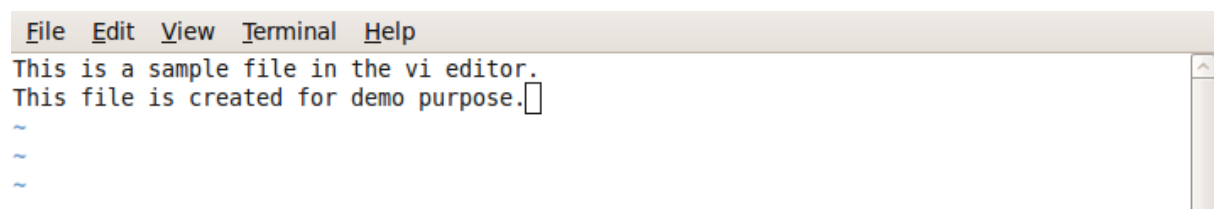
Snapshot (3)

```
File  Edit  View  Terminal  Help
This is a sample file in the vi editor.
This file is created for demo purpose.
~
~
~
~
```

Figure 4.1.3

Saving and Quitting Vi

:q – To quit without saving, however vi editor will warn you for this and not let you quit.

:q! – To quit without saving. It lets quit Vi without saving.

:w - In command mode, type :w to save data.

:wq – we can combine w with q for saving and quitting.

:zz - we can also press z key twice to Save and quit same like

:wq. ->Output with **:q** shown in below snapshot:

Snapshot (4)

```
File  Edit  View  Terminal  Help
This is a sample file in the vi editor.
This file is created for demo purpose.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
E37: No write since last change (add ! to override)
```
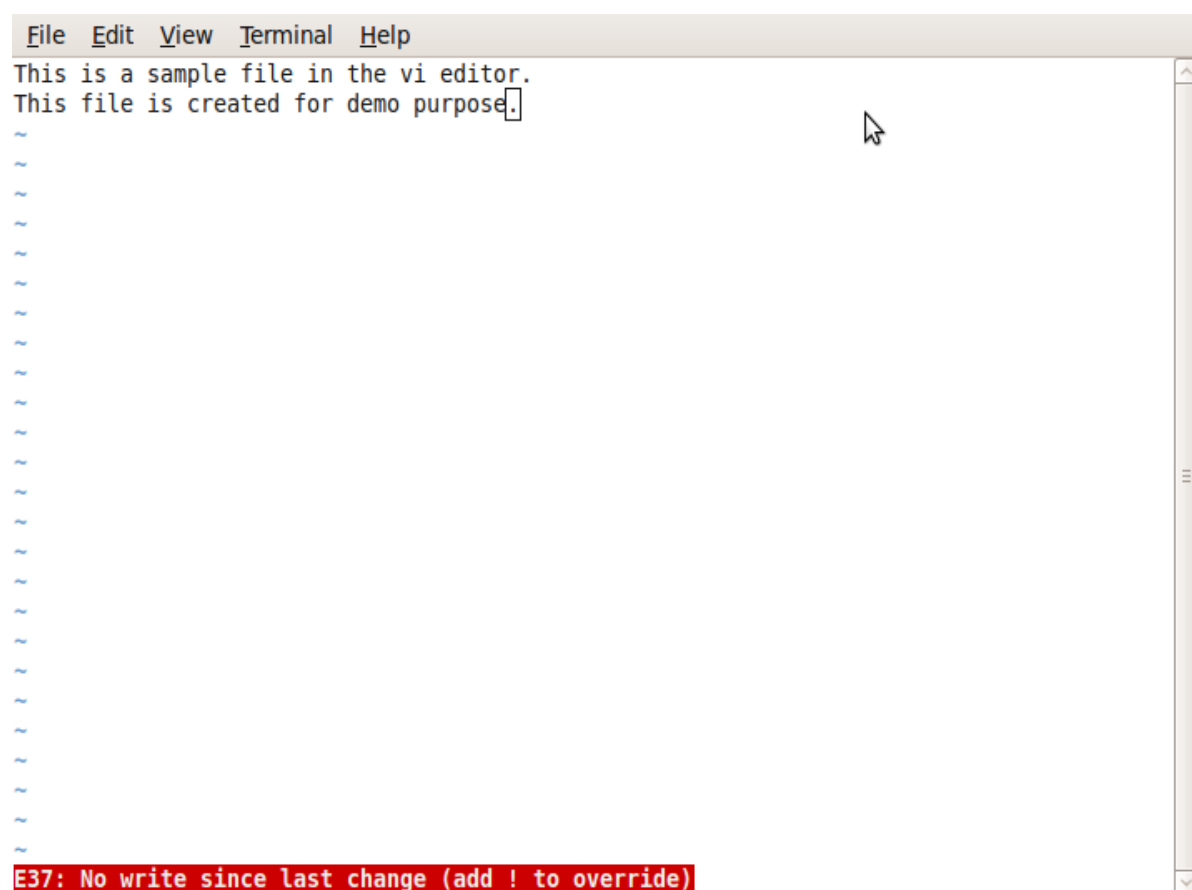
Figure 4.1.4

Note: File would not be saved, flashed warning and Vi editor would not terminate.

->Output with **:q!** shown in below snapshot:

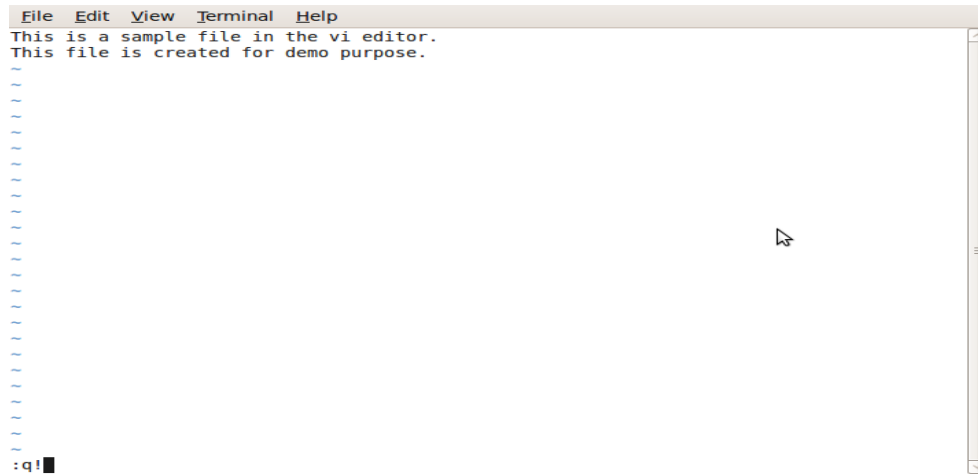**Snapshot (5)**

Figure 4.1.5

Note: File would not be saved and vi editor would also be terminated

->Output with **: w** shown in below snapshot:
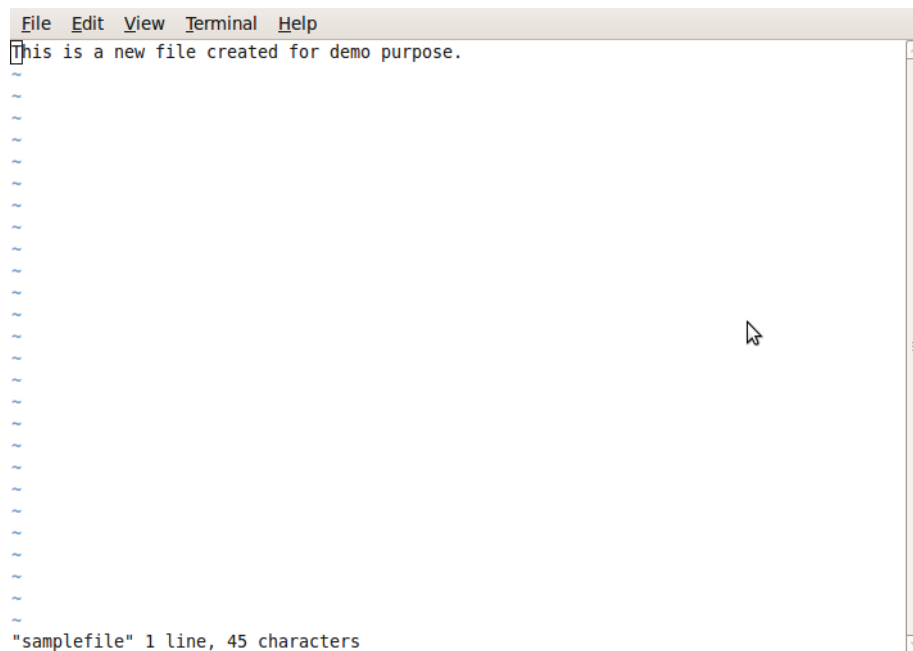
Snapshot (6)

Figure 4.1.6

Note: Data would be saved without terminating vi editor

->Output with **:wq** shown in below snapshot:

Snapshot (7)

Figure 4.1.7

Note: File would be saved and vi editor would also be terminated.

Moving Cursor without affecting your text in Command Mode

In command mode, press following keys for moving cursor:

Command	Function
K	Moves the cursor one line up
J	Moves the cursor one line down
H	Moves the cursor left one char
L	Moves the cursor right one char

Note: You can enter the number of times you want command to be executed like 2k will move the cursor 2 lines up.

Similarly, some other commands for cursor movement in Command Mode

Command	Function
:n	Move to the nth line of the file
L	Move to the bottom of the screen
H	Move to the Top of the screen
G	Move to the last line of the file
W	Position cursor to the next word
B	Position cursor to the previous word

Editing Files in the Insert Mode

After pressing below keys you would able to edit file.

Command	Function
i	Insert text to the left of the cursor
I	Add text in the begging of the current line
a	Append text to the right of cursor
A	Append text at the end of line
o	Creates a new line below
O	Creates a new line above

Deleting Characters in the Command Mode

Command	Function
X	Deletes the char under the cursor location
X	Deletes the char before the cursor location
dw	Deletes from the current cursor location to the next word
d^	Deletes from the current cursor position to the beginning of the line
d$	Deletes from the current cursor position to the end of the line
dd	Deletes the line the cursor is on.

4.1.5 Deleting and inserting lines

Command	Function
dd	Deletes the line the cursor is on.
o	Creates a new line below
O	Creates a new line above

4.1.6 Deleting and replacing character

Command	Function
R	Replaces the character under the cursor, the vi returns to command mode after replacement is entered
R	Overwrite multiple characters beginning with the character currently under the cursor, press ESC to stop overwriting
S	Replaces the current char with the char you type. After that you left in insert mode
S	Deletes the line the cursor is on and replaces with the new text. After the nex text is entered vi remains in the insert mode.

4.1.7 Searching for Strings

Vi supports two types of searching char or string searching.

String searching – For string searching / and ? are used.

- / - For searching forward, downwards in the file.
- ?- For searching backward, upwards in the file.

The n and N commands repeat the previous search commands in the same or opposite direction.

Important: Search and Replace (:s)

: address/source_pattern/target_pattern/flags

Example:

:1,$s/TEXT/UNIX/g

Address: 1,$ addresses all lines in a file

Flag: g is very common flag which carried out substitution throughout the file. Otherwise it replaces only the first occurrence.

Other example:

: 1,30s/TEXT/UNIX/g

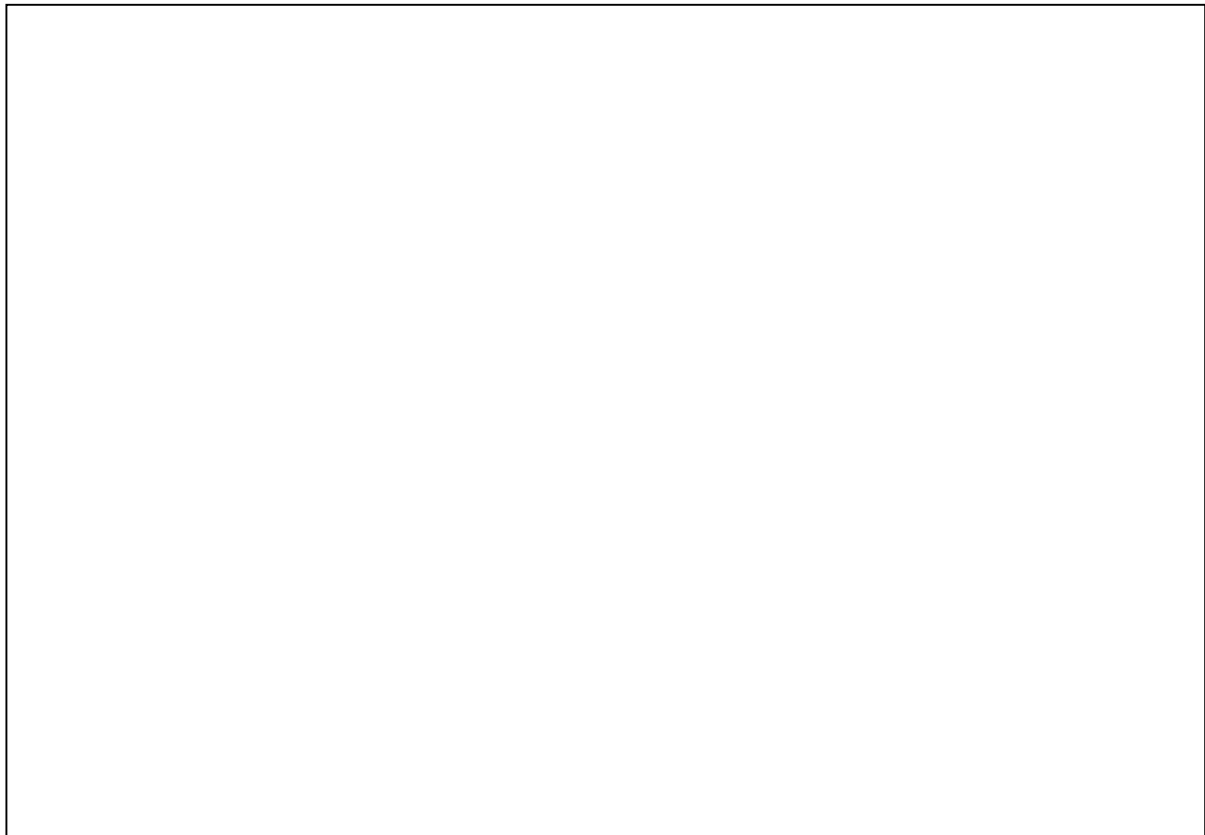It would carry the replacement only in the first 30 lines.

Figure 4.1.8

```
File  Edit  View  Terminal  Help
This is a new file created for demo purpose.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:1,$s/file/directory/g█
```

```
File  Edit  View  Terminal  Help
This is a new directory created for demo purpose.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:1,$s/file/directory/g
```
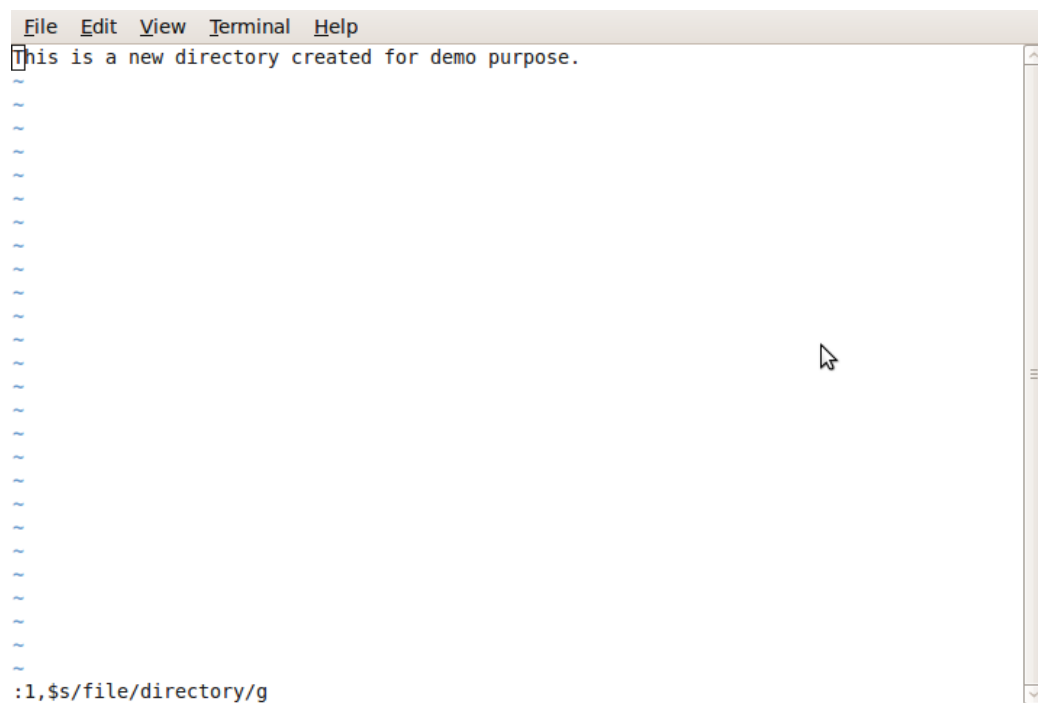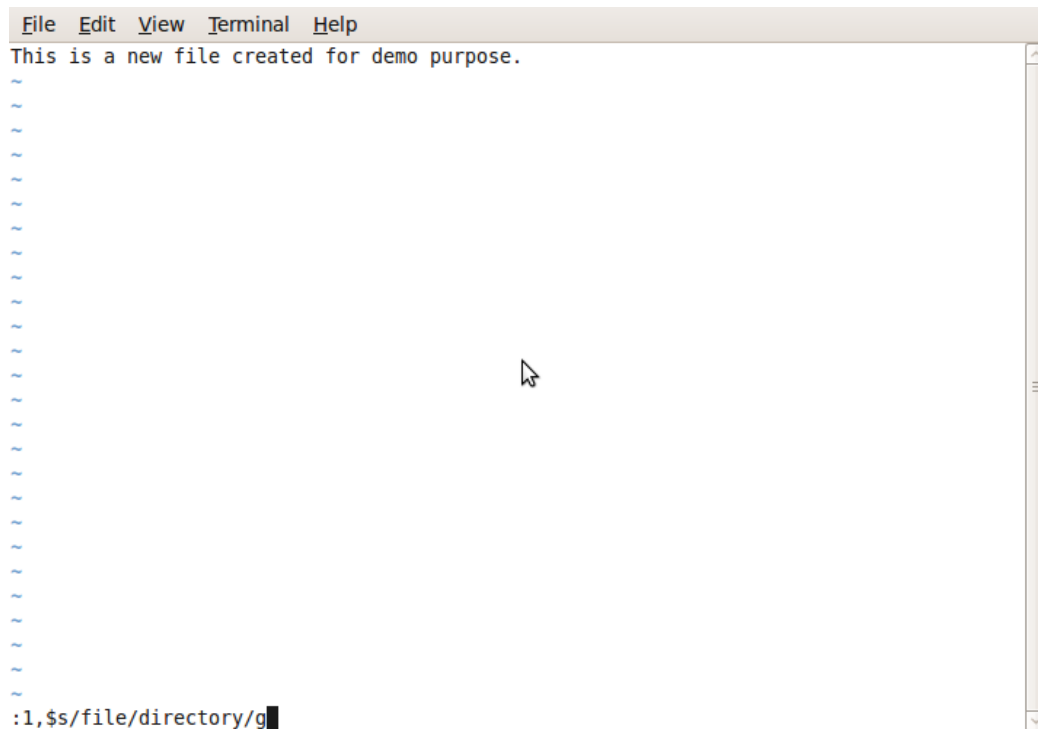
Figure 4.1.9

4.1.8 Yanking

Yanking means copying data.

Copy and Paste Commands

Command	Function
yy	Yanking or Copying the current line
yw	Copies the current word
p	Puts the copied text after the cursor
P	Puts the yanked text before the cursor

File   Edit   View   Terminal   Help
```
This is a new directory created for demo purpose.
This is a new directory created for demo purpose.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
```
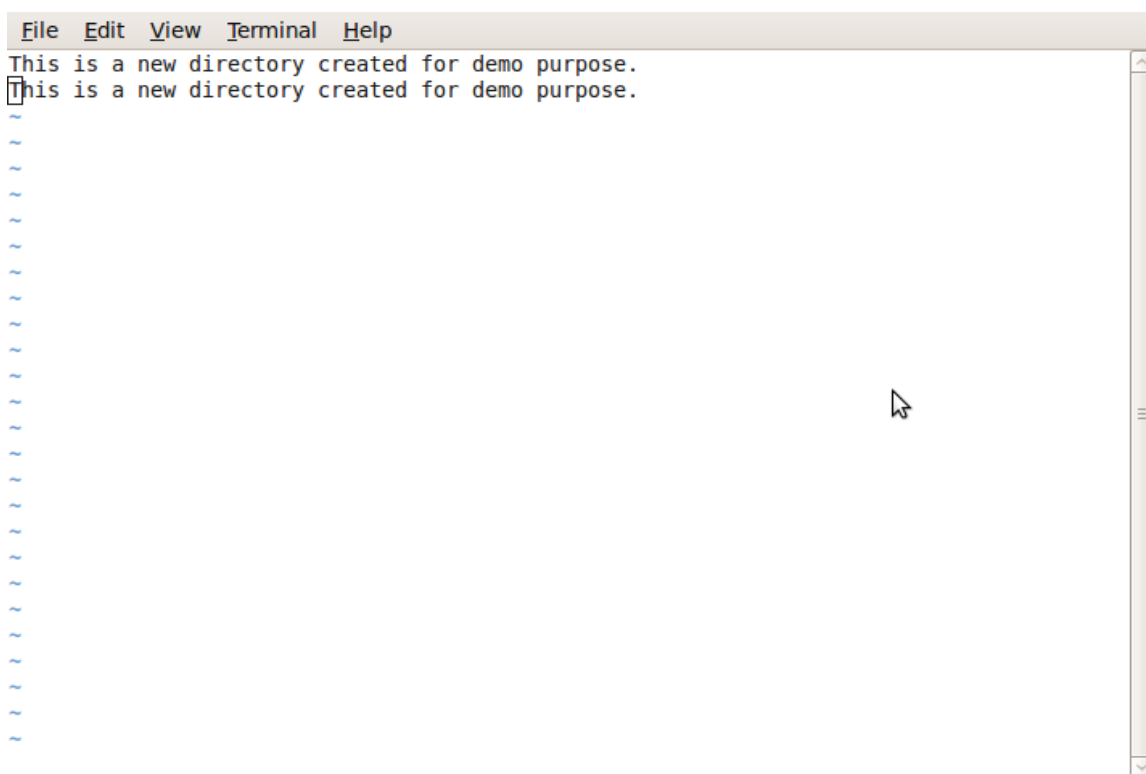
Figure 4.1.10

When a process executes a system call, the execution mode of the process changes from user mode to kernel mode.

Self-assessment Questions

1) Which one of the following statement is not true?

   a) vi editor is the improved version of vi editor

   b) vi editor commands are not case sensitive

   c) vi editor has two modes of operation: command mode and insert mode

   d) vi stands for visual editor

2) Which command is used to close the vi editor?

   a) q

   b) wq

   c) both (a) and (b)

   d) none of the mentioned

3) In vi editor, the key combination CTRL+f

   a) Moves screen down one page

   b) Moves screen up one page

   c) Moves screen up one line

   d) Moves screen down one line

4) Which vi editor command copies the current line of the file?

   a) yy

   b) yw

   c) yc

   d) none of the mentioned

5) Which command is used to close the vi editor?

   a) / or ?

   b) f or F

   c) t or T

   d) none of the mentioned

6) In vi editor, which command reads the content of another file

   a) read

   b) r

   c) ex

   d) none of the mentioned

7) vi editor commands are not case sensitive.

   a) True

   b) False

8) Which command is used to delete the character before the cursor location in vi editor?

   a) X

   b) x

   c) D

   d) d

9) Which one of the following statement is true?
   a) Auto indentation is not possible in vi editor
   b) Auto indentation can be set using the command ':set ai'
   c) Auto indentation can be set using the command ':set noai'
   d) Auto indentation is set by default in vi editor

10) Which command searches the string in file opened in vi editor?
   a) / or ?   b) f or F
   c) t or T   d) none of the mentioned

☰ Summary

○ Vi operates in three modes. The command mode is used to enter commands that operate on text or control cursor motion. The input mode is used to enter text. The ex mode is used for file handling and substitution.

○ Navigation is performed in the command mode. You can move in four directions(h,j,k and l) or move along a line, using a word as navigation unit

○ Editing functions are also performed in command mode. You can delete characters (x and X) and lines(dd) and yank or copy lines(yy).

○ Vi supports two types of searching char or string searching. For string searching / and are used.

Terminal Questions

1. Define the concept of text processing in vi editor. Also explain the different modes in vi editor.

2. Illustrate the procedure to switch from one mode to another mode in vi editor.

3. Illustrate how to search and replace a string in vi editor

4. Explain the process of yanking

5. Demonstrate following functionalities in vi editor:

   - How to delete and insert line in a file?

   - How to delete and replace character in a file?

Answer Keys

Self-assessment Questions	
Question No.	Answer
1	b
2	c
3	a
4	a
5	a
6	c
7	b
8	a
9	b
10	a

Activity

Activity Type: Online/Offline   **Duration: 30 Minutes**

Description:

Enter the following text into a file named intro.txt and perform the operations indicated below:

In computing electronic data, the term text processing is a mechanism of creation or manipulation of data. Texts are commonly referred to the alphanumeric characters entered on the keyboard by the user who is manipulating or creating the text, but text processing here uses standard formats for character encoding of the required text.

Shell programs or any other documents which are entered and edited on the computer using a specific software package called the editor. There are many types of editors available on UNIX systems such as vi (visual editor), vim (vi improved version), emacs and pico.

This chapter describes the text processing on vi editor. Text processing in vi editor involves computer commands which invokes content, changes the content and cursor movement.

a) Add line numbers
b) Change all the occurrences of the word TEXT to UNIX
c) Swap first and second paragraph
d) Join line number 3, 4 , and 5
e) Delete line number 7
f) Yank the very first line of the text.
g) Enter your number at the end of the file.
h) Save the file.

Bibliography

📖 e-References

- This website was referred on 3rd May 2016, while developing content for vi editor
  http://www.emerson.emory.edu/services/editors/vi/vi.html

- This website was referred on 3rd May 2016, while developing content for vi editor
  http://www.oualline.com/vim/10/top_10.html

- This website was referred on 3rd May 2016, while developing content for vi editor
  https://www.cs.colostate.edu/helpdocs/vi.html

- This website was referred on 3rd May 2016, while developing content for vi editor
  http://www.linfo.org/vi/search.html

- This website was referred on 3rd May 2016, while developing content for vi editor
  http://www.computerhope.com/unix/uvi.htm

- This website was referred on 3rd May 2016, while developing content for vi editor
  http://www.dba-oracle.com/linux/t_linux_54_cutting_copying_pasting.htm

📕 External Resources

- Maurice J. Bach, The Design of Unix Operating System, (2010) Pearson Education
- S. Prata, Advance UNIX, a Programmer's Guide, (2011), BPB Publications, and New Delhi,
- B.W. Kernighan & R. Pike, The UNIX Programming Environment, (2009) Prentice Hall of India.
- Jack Dent Tony Gaddis, Guide to UNIX Using LINUX, (2010) Vikas/ Thomson Pub. House Pvt. Ltd.

📹 Video Links

Topic	Link
Vi Commands and Usage	https://www.youtube.com/watch?v=TBu6qxd5uAc

Notes: