



DHA Suffa University
CS 206 – Operating Systems - Lab
Spring 2017



Lab 02 – Visual Editor in Linux and Running C++

Objective(s):

- Installing C++ Compiler in Linux
- Compiling and Running C++ code in Linux
- To edit text in Visual Editor
- To learn about command and insertion mode
- To learn about text navigation commands

Installing C++ Compiler for Linux:

In order to run a C++ program in Linux terminal, we need to install G++ compiling configuration in our system.

The installation procedure works as follows:

1. Update packages by running the following command:

sudo apt-get update

2. Install g++ by running the following command:

sudo apt-get install g++

Note: You need to be logged in as administrator to run all sudo commands.

- Logged in as guest user will prevent you from installing or updating any software.
- The system will refuse to run those commands by showing the following (or similar to following) error:

unable to change root gid: Operation not permitted

Modes:

Vi has two modes insertion mode and command mode. The editor begins in command mode, where the cursor movement and text deletion and pasting occur. Insertion mode begins upon entering an insertion or change command. [ESC] returns the editor to command mode (where you can quit, for example by typing :q!). Most commands execute as soon as you type them except for "colon" commands which execute when you press the return key.

Motion

h	Move left
j	Move down
k	Move up
l	Move right
w	Move to next word
W	Move to next blank delimited word
b	Move to the beginning of the word
B	Move to the beginning of blank delimited word
e	Move to the end of the word
E	Move to the end of Blank delimited word
(Move a sentence back
)	Move a sentence forward
{	Move a paragraph back
}	Move a paragraph forward
0	Move to the begining of the line
\$	Move to the end of the line
1G	Move to the first line of the file
G	Move to the last line of the file
nG	Move to nth line of the file
:n	Move to nth line of the file
fc	Move forward to c
Fc	Move back to c
H	Move to top of screen
M	Move to middle of screen
L	Move to botton of screen
%	Move to associated (), { }, []

Quitting

:x	Exit, saving changes
:q	Exit as long as there have been no changes
ZZ	Exit and save changes if any have been made
:q!	Exit and ignore any changes

Inserting Text

i	Insert before cursor
I	Insert before line
a	Append after cursor
A	Append after line
o	Open a new line after current line
O	Open a new line before current line
r	Replace one character
R	Replace many characters

Deleting Text

Almost all deletion commands are performed by typing d followed by a motion. For example, dw deletes a word. A few other deletes are:

x	Delete character to the right of cursor
X	Delete character to the left of cursor
D	Delete to the end of the line
dd	Delete current line
:d	Delete current line

Files

:w <i>file</i>	Write to <i>file</i>
:r <i>file</i>	Read <i>file</i> in after line
:n	Go to next file
:p	Go to previos file
:e <i>file</i>	Edit <i>file</i>
!! <i>program</i>	Replace line with output from <i>program</i>

Changing text

The change command is a deletion command that leaves the editor in insert mode. It is performed by typing `c` followed by a motion. For example `cw` changes a word. A few other change commands are:

C	Change to the end of the line
cc	Change the whole line

Putting text

p	Put after the position or after the line
P	Put before the position or before the line

Search for strings

/string	Search forward for <i>string</i>
?string	Search back for <i>string</i>
n	Search for next instance of <i>string</i>
N	Search for previous instance of <i>string</i>

Replace

The search and replace function is accomplished with the `:s` command. It is commonly used in combination with ranges or the `:g` command (below).

:s/pattern/string/flags	Replace <i>pattern</i> with <i>string</i> according to <i>flags</i> .
g	Flag - Replace all occurrences of pattern
c	Flag - Confirm replaces.
&	Repeat last <code>:s</code> command

Other

~	Toggle upper and lower case
J	Join lines
.	Repeat last text-changing command
u	Undo last change
U	Undo all changes to line

Lab Task:

Write a C++ code using Vi (Visual Editor) in Linux terminal. Use the first part of this lab manual to edit the file.

Compiling a C++ file:

To compile a C++ file:

1. You need to be navigated into the directory that resides the file to be compiled.
2. You need to run the following command:

g++ -c filename.cpp

- where *filename* is the name of the file you want to compile and *cpp* is the extension of that file.
- This command will create an object file of the same name as the *cpp* file. (i.e. *filename.o*)
- **Note:** Remember all commands in Linux terminal are case-sensitive.

Linking the Object file:

To execute the code, object file needs to be linked. In order to do so, we need to run the following command.

g++ -o filename filename.cpp

- This command will link the object file and make an executable file by the same name as you file (or the name you give after writing -o)
- You can also run this command without first compiling the code.

Running the executable file:

To execute the code, you need to only write the file name preceded by *./* without the quotations.

./filename