电子科技大学

计算机科学与工程学院

标准实验报告

（实验）课程名称 Unix 操作系统

电子科技大学教务处制表

电子科技大学

电子科技大学

实验报告

学生姓名：刘芷溢　　　　学号：2020080907009

指导教师：聂晓文　　　　实验地点：主楼 A2-412

实验时间：2022/3/25

一、实验室名称：计算机学院实验中心

二、实验项目名称：Unix 基本操作与 Vi 编辑器

三、实验学时：4 学时

四、实验原理：Unix 操作系统和 Vi 编辑器

五、实验目的：

在实验中验证 Unix 的基本命令，熟练掌握 vi 编辑器。

六、实验内容：

1. Unix 基本操作
    a) 登录、退出、重启、关机，根据现象，解释登录过程与退出过程
    b) Ubuntu 安装程序：apt install sudo、apt install vim
    c) 基本命令
        i. date、cal、passwd、who
        ii. man，man 的分卷、分节
        iii. unix 目录结构
        iv. 绝对路径与相对路径
        v. mkdir、rmdir
        vi. cd 命令，cd，cd 绝对路径，cd 相对路径，cd ~，cd －，cd 父目录
        vii. ls 命令，ls －al，解释 ls －al /dev 的输出
        viii. rm －fr
2. Vi 编辑器
    a) 编辑文件的方式，退出
    b) :help 命令
    c) 移动光标
    d) Vi 的模式：命令模式、输入模式、虚拟模式、命令行模式
    e) dd、x、p
    f) 快速退出、切换 shell 的技巧(ZZ)
    g) 编辑多个文件的技巧，利用 file、buffer、window 概念解释

h） 查看寄存器，解释有哪些寄存器

i） 观察 motion 与 range

j） 寄存器组合命令，观察寄存器变化

k） 选项，showmode、number

l） 连接两行 J

七、实验器材（设备、元器件）：

PC 微机一台

八、实验步骤：

1.下载 VMware 虚拟机

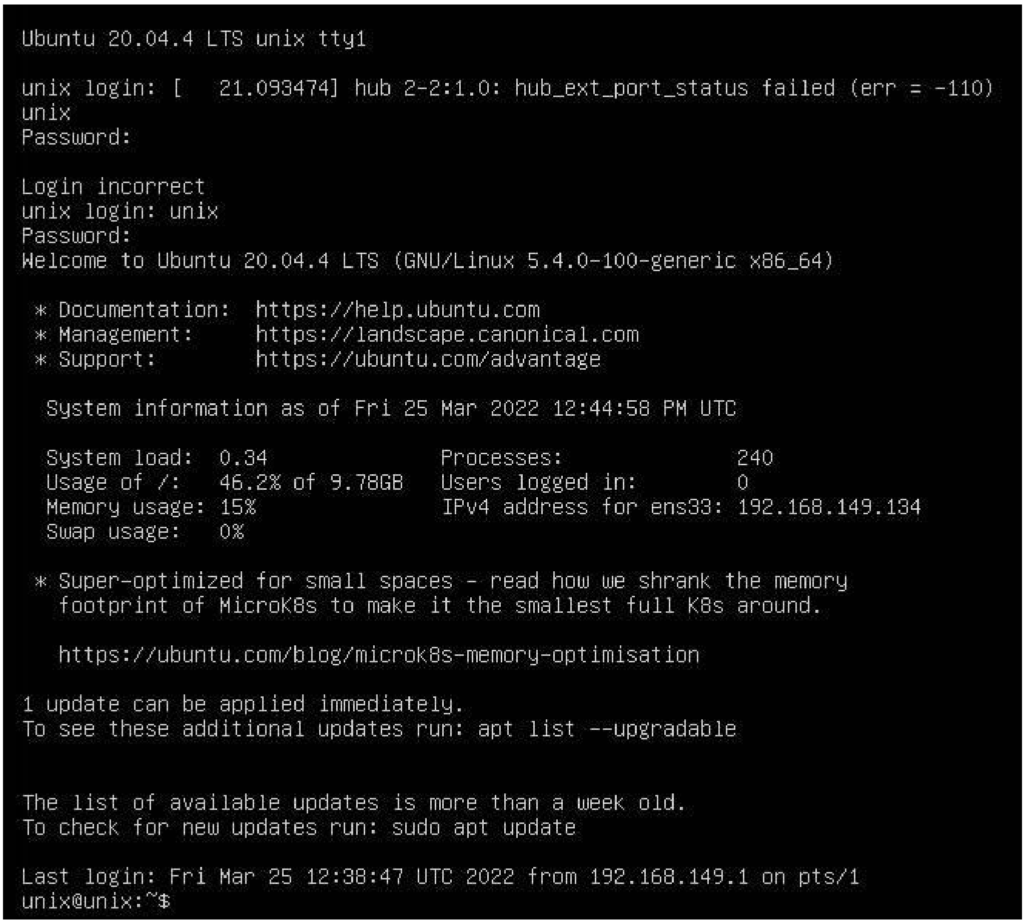
2.在虚拟机上安装 Ubuntu 服务器版本，进行环境配置

3.执行实验要求的各项操作

九、实验数据及结果分析：

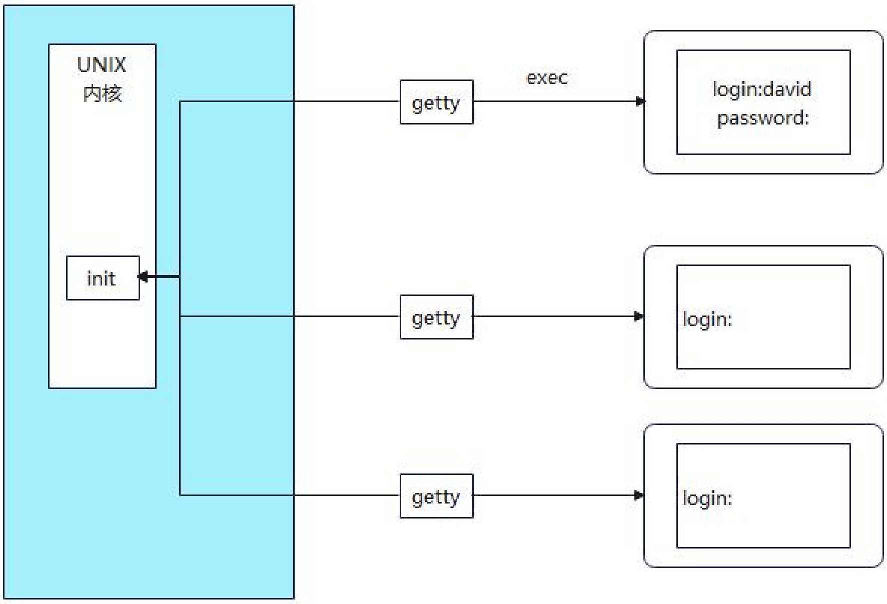
1.

a)登录：

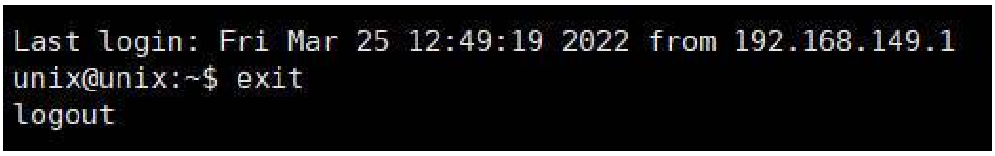


```
Ubuntu 20.04.4 LTS unix tty1

unix login: [   21.093474] hub 2-2:1.0: hub_ext_port_status failed (err = -110)
unix
Password:

Login incorrect
unix login: unix
Password:
Welcome to Ubuntu 20.04.4 LTS (GNU/Linux 5.4.0-100-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Fri 25 Mar 2022 12:44:58 PM UTC

  System load:  0.34              Processes:             240
  Usage of /:   46.2% of 9.78GB   Users logged in:       0
  Memory usage: 15%               IPv4 address for ens33: 192.168.149.134
  Swap usage:   0%

 * Super-optimized for small spaces - read how we shrank the memory
   footprint of MicroK8s to make it the smallest full K8s around.

   https://ubuntu.com/blog/microk8s-memory-optimisation

1 update can be applied immediately.
To see these additional updates run: apt list --upgradable


The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Fri Mar 25 12:38:47 UTC 2022 from 192.168.149.1 on pts/1
unix@unix:~$
```

登录过程：init 进程 fork 出子进程 getty，执行登陆程序；

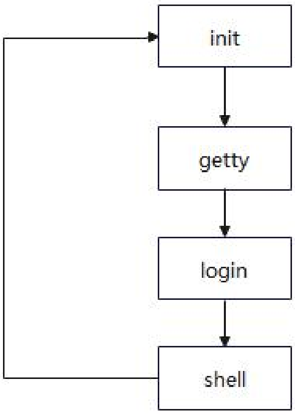


退出：



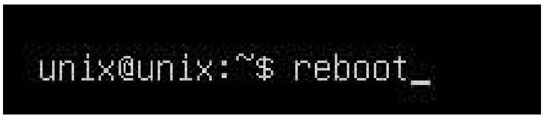
```
Last login: Fri Mar 25 12:49:19 2022 from 192.168.149.1
unix@unix:~$ exit
logout
```

退出过程：将进程销毁，重新回到 init 进程；



登录与退出循环

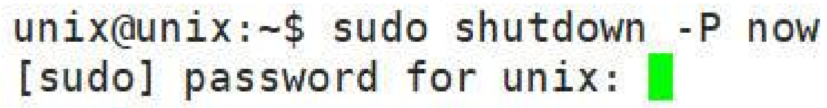
重启：



```
unix@unix:~$ reboot_
```

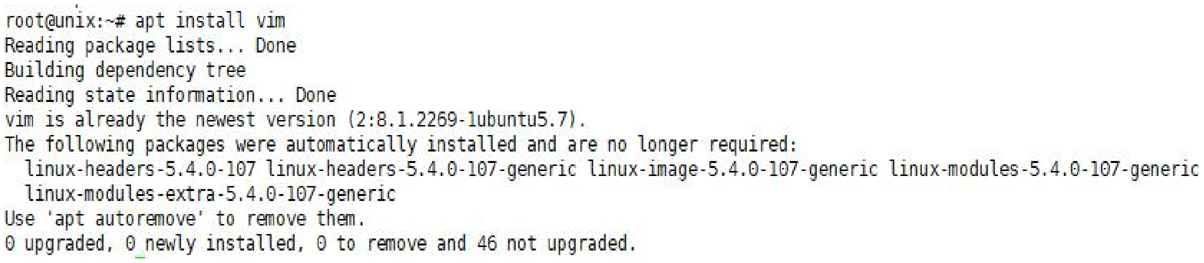
关机：

输入密码后关掉所有进程后关机；



```
unix@unix:~$ sudo shutdown -P now
[sudo] password for unix:
```

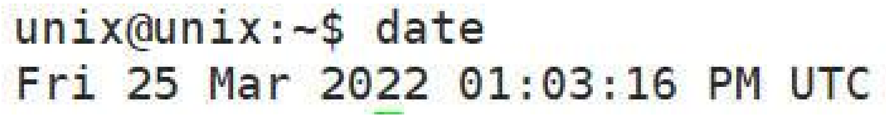
B）：Ubuntu 安装程序：



```
root@unix:~# apt install vim
Reading package lists... Done
Building dependency tree
Reading state information... Done
vim is already the newest version (2:8.1.2269-1ubuntu5.7).
The following packages were automatically installed and are no longer required:
  linux-headers-5.4.0-107 linux-headers-5.4.0-107-generic linux-image-5.4.0-107-generic linux-modules-5.4.0-107-generic
  linux-modules-extra-5.4.0-107-generic
Use 'apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 46 not upgraded.
```

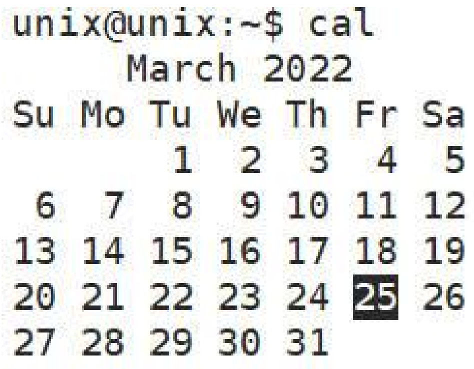
C）：基本命令

Date 命令：



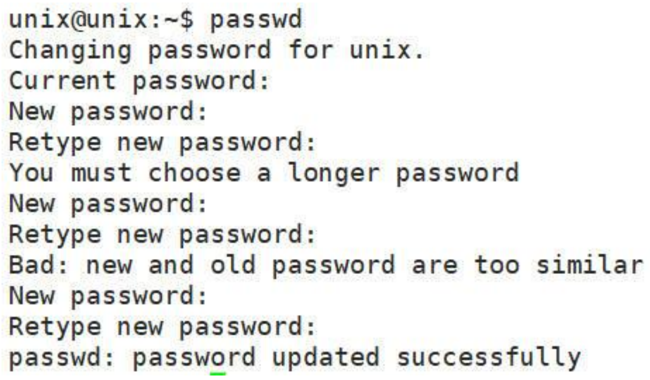
```
unix@unix:~$ date
Fri 25 Mar 2022 01:03:16 PM UTC
```

Cal 命令：



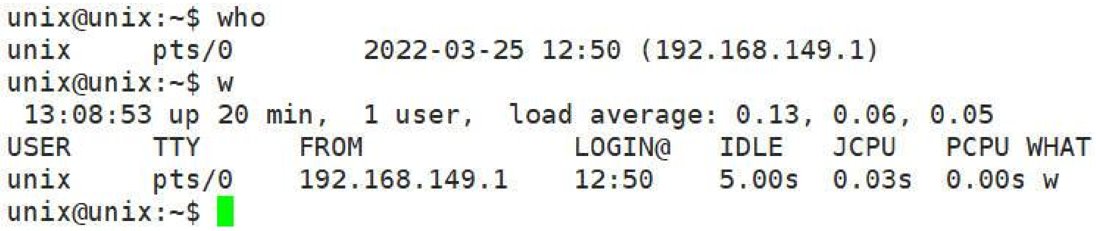
```
unix@unix:~$ cal
     March 2022
Su Mo Tu We Th Fr Sa
       1  2  3  4  5
 6  7  8  9 10 11 12
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28 29 30 31
```

Passwd 命令：



```
unix@unix:~$ passwd
Changing password for unix.
Current password:
New password:
Retype new password:
You must choose a longer password
New password:
Retype new password:
Bad: new and old password are too similar
New password:
Retype new password:
passwd: password updated successfully
```

Who 命令：



```
unix@unix:~$ who
unix     pts/0        2022-03-25 12:50 (192.168.149.1)
unix@unix:~$ w
 13:08:53 up 20 min,  1 user,  load average: 0.13, 0.06, 0.05
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
unix     pts/0    192.168.149.1    12:50    5.00s  0.03s  0.00s w
unix@unix:~$
```

Man 命令：

1、Standard commands (标准命令)

2、System calls (系统调用)

3、Library functions (库函数)

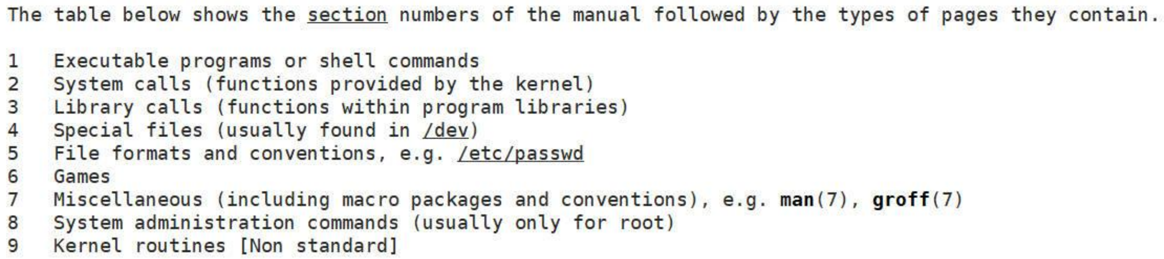
4、Special devices (设备说明)

5、File formats (文件格式)

6、Games and toys (游戏和娱乐)

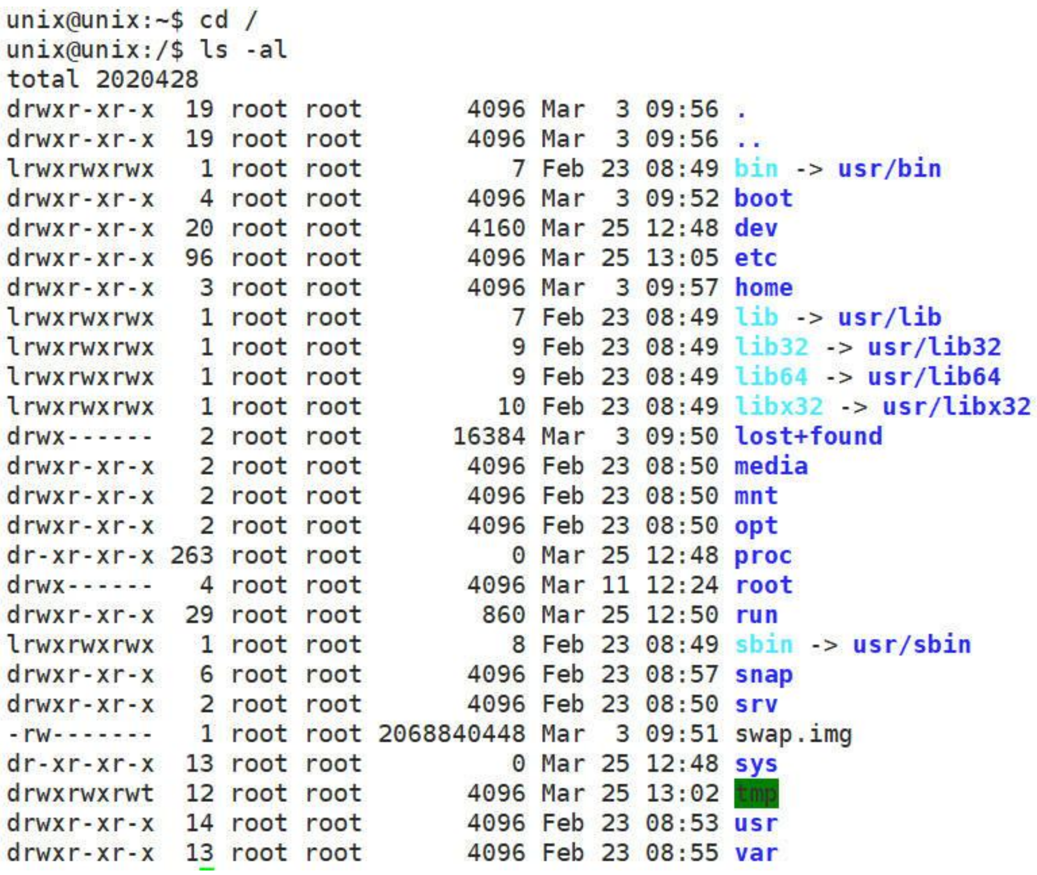
7、Miscellaneous (杂项)

8、Administrative Commands (管理员命令)



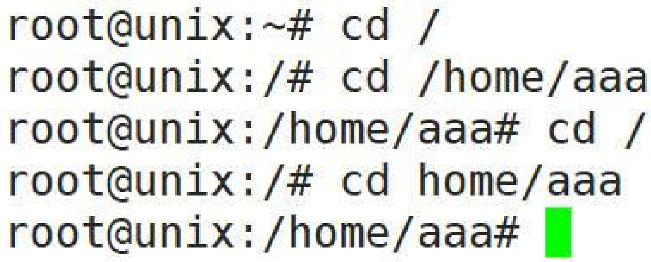
```
The table below shows the section numbers of the manual followed by the types of pages they contain.

1   Executable programs or shell commands
2   System calls (functions provided by the kernel)
3   Library calls (functions within program libraries)
4   Special files (usually found in /dev)
5   File formats and conventions, e.g. /etc/passwd
6   Games
7   Miscellaneous (including macro packages and conventions), e.g. man(7), groff(7)
8   System administration commands (usually only for root)
9   Kernel routines [Non standard]
```

Unix 目录结构：



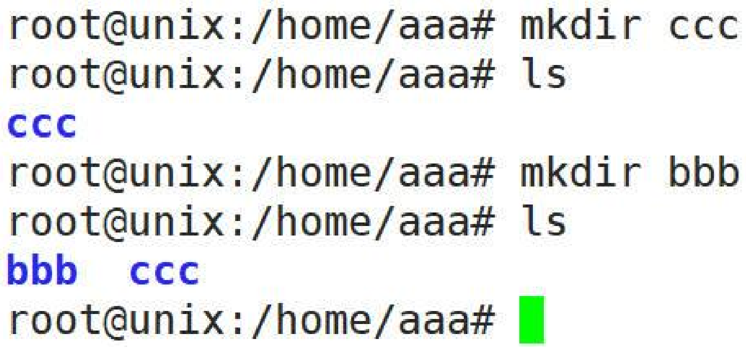
```
unix@unix:~$ cd /
unix@unix:/$ ls -al
total 2020428
drwxr-xr-x  19 root root       4096 Mar  3 09:56 .
drwxr-xr-x  19 root root       4096 Mar  3 09:56 ..
lrwxrwxrwx   1 root root          7 Feb 23 08:49 bin -> usr/bin
drwxr-xr-x   4 root root       4096 Mar  3 09:52 boot
drwxr-xr-x  20 root root       4160 Mar 25 12:48 dev
drwxr-xr-x  96 root root       4096 Mar 25 13:05 etc
drwxr-xr-x   3 root root       4096 Mar  3 09:57 home
lrwxrwxrwx   1 root root          7 Feb 23 08:49 lib -> usr/lib
lrwxrwxrwx   1 root root          9 Feb 23 08:49 lib32 -> usr/lib32
lrwxrwxrwx   1 root root          9 Feb 23 08:49 lib64 -> usr/lib64
lrwxrwxrwx   1 root root         10 Feb 23 08:49 libx32 -> usr/libx32
drwx------   2 root root      16384 Mar  3 09:50 lost+found
drwxr-xr-x   2 root root       4096 Feb 23 08:50 media
drwxr-xr-x   2 root root       4096 Feb 23 08:50 mnt
drwxr-xr-x   2 root root       4096 Feb 23 08:50 opt
dr-xr-xr-x 263 root root          0 Mar 25 12:48 proc
drwx------   4 root root       4096 Mar 11 12:24 root
drwxr-xr-x  29 root root        860 Mar 25 12:50 run
lrwxrwxrwx   1 root root          8 Feb 23 08:49 sbin -> usr/sbin
drwxr-xr-x   6 root root       4096 Feb 23 08:57 snap
drwxr-xr-x   2 root root       4096 Feb 23 08:50 srv
-rw-------   1 root root 2068840448 Mar  3 09:51 swap.img
dr-xr-xr-x  13 root root          0 Mar 25 12:48 sys
drwxrwxrwt  12 root root       4096 Mar 25 13:02 tmp
drwxr-xr-x  14 root root       4096 Feb 23 08:53 usr
drwxr-xr-x  13 root root       4096 Feb 23 08:55 var
```

绝对路径与相对路径：



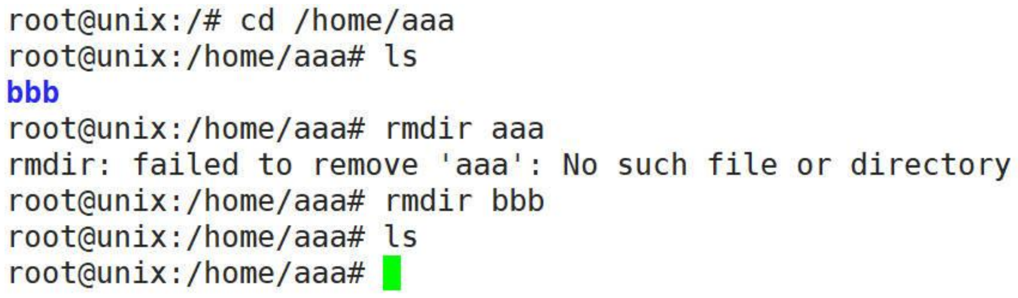
```
root@unix:~# cd /
root@unix:/# cd /home/aaa
root@unix:/home/aaa# cd /
root@unix:/# cd home/aaa
root@unix:/home/aaa# 
```

Mkdir 命令：



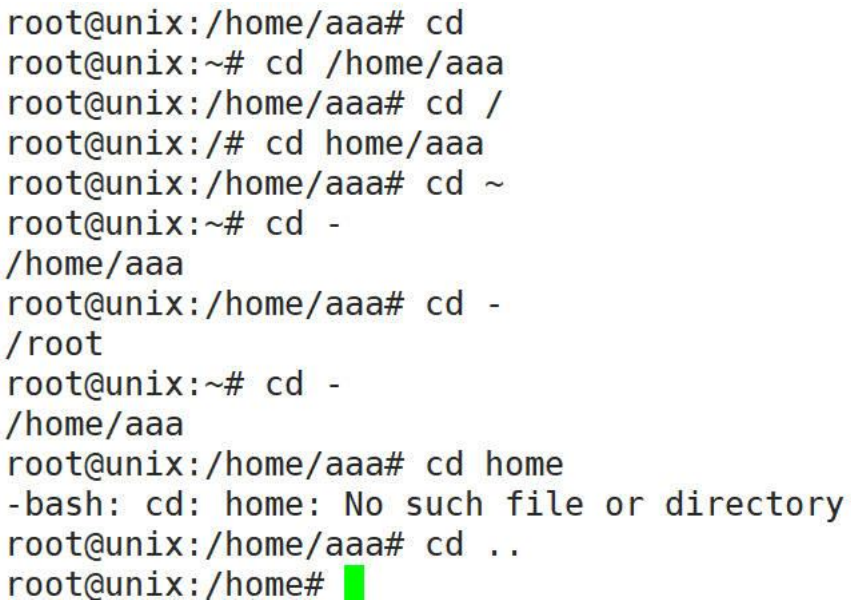
```
root@unix:/home/aaa# mkdir ccc
root@unix:/home/aaa# ls
ccc
root@unix:/home/aaa# mkdir bbb
root@unix:/home/aaa# ls
bbb  ccc
root@unix:/home/aaa# 
```

Rmdir 命令：



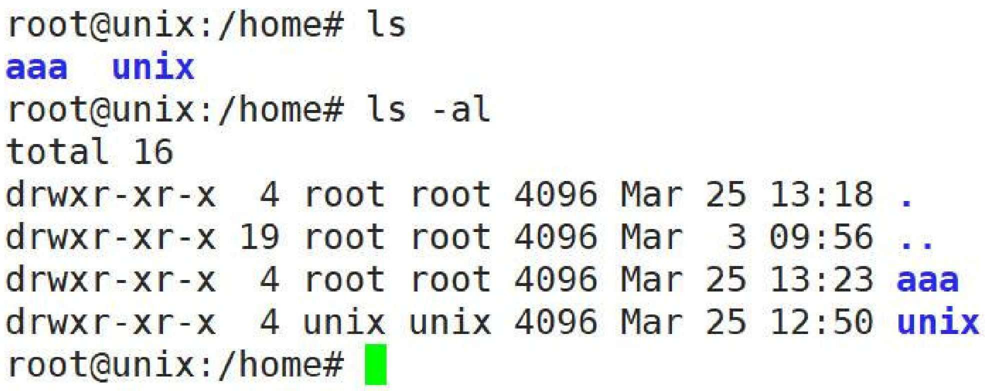
```
root@unix:/# cd /home/aaa
root@unix:/home/aaa# ls
bbb
root@unix:/home/aaa# rmdir aaa
rmdir: failed to remove 'aaa': No such file or directory
root@unix:/home/aaa# rmdir bbb
root@unix:/home/aaa# ls
root@unix:/home/aaa# 
```

Cd 命令：



```
root@unix:/home/aaa# cd
root@unix:~# cd /home/aaa
root@unix:/home/aaa# cd /
root@unix:/# cd home/aaa
root@unix:/home/aaa# cd ~
root@unix:~# cd -
/home/aaa
root@unix:/home/aaa# cd -
/root
root@unix:~# cd -
/home/aaa
root@unix:/home/aaa# cd home
-bash: cd: home: No such file or directory
root@unix:/home/aaa# cd ..
root@unix:/home# 
```

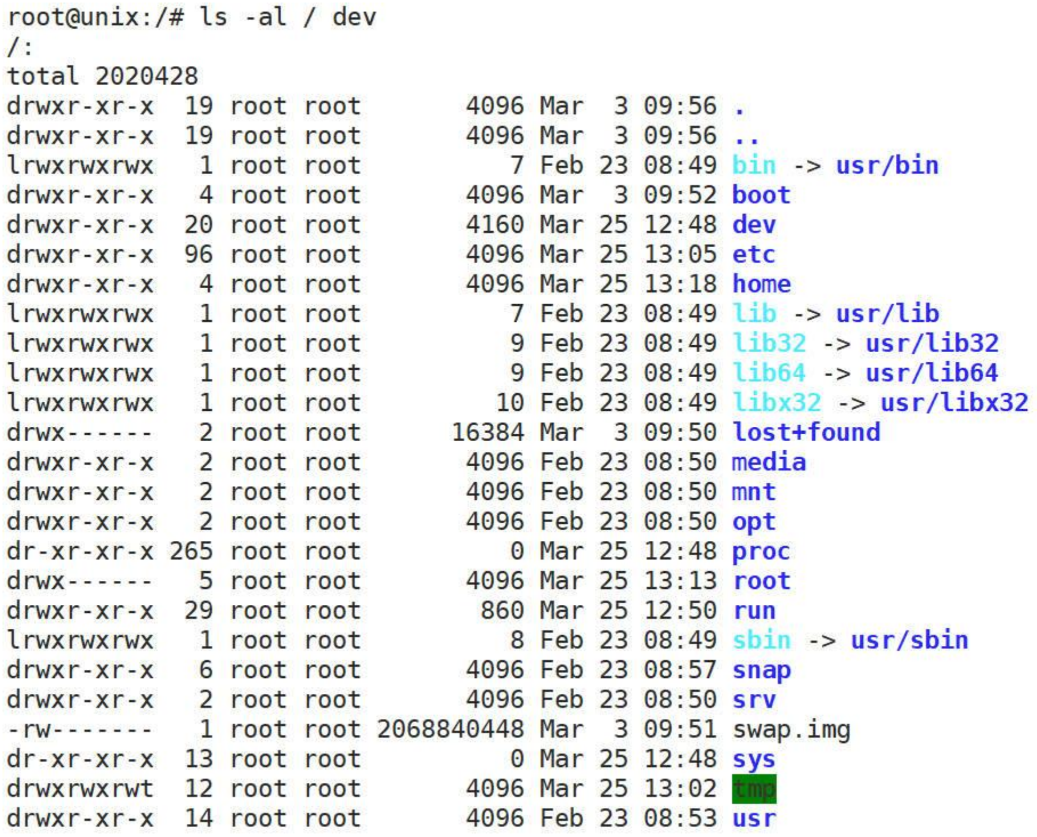
Ls 命令：



```
root@unix:/home# ls
aaa  unix
root@unix:/home# ls -al
total 16
drwxr-xr-x  4 root root 4096 Mar 25 13:18 .
drwxr-xr-x 19 root root 4096 Mar  3 09:56 ..
drwxr-xr-x  4 root root 4096 Mar 25 13:23 aaa
drwxr-xr-x  4 unix unix 4096 Mar 25 12:50 unix
root@unix:/home# 
```

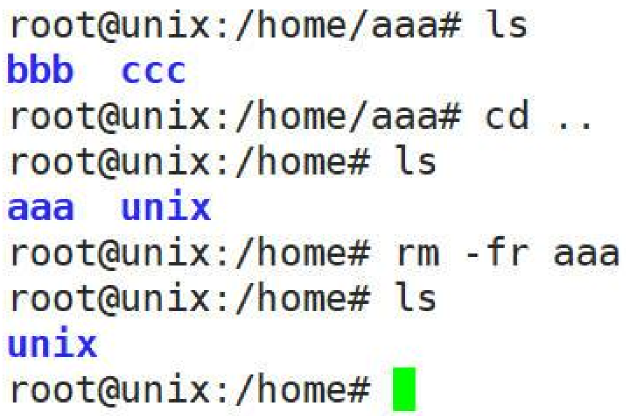
Ls -al / dev：显示/dev 目录下所有文件的详细信息，以长格式列出文件，显示文件的详细信息时，第一列有 10 个字符，第一个字符表示文件类型，其他 9

个字符每 3 个一组表示权限；第二列表示链接数；第三列表示文件所有者；第四列显示用户组；第五列显示文件大小；第 6 列显示上一次修改的日期和时间；第七列为文件名；



```
root@unix:/# ls -al / dev
/:
total 2020428
drwxr-xr-x  19 root root       4096 Mar  3 09:56 .
drwxr-xr-x  19 root root       4096 Mar  3 09:56 ..
lrwxrwxrwx   1 root root          7 Feb 23 08:49 bin -> usr/bin
drwxr-xr-x   4 root root       4096 Mar  3 09:52 boot
drwxr-xr-x  20 root root       4160 Mar 25 12:48 dev
drwxr-xr-x  96 root root       4096 Mar 25 13:05 etc
drwxr-xr-x   4 root root       4096 Mar 25 13:18 home
lrwxrwxrwx   1 root root          7 Feb 23 08:49 lib -> usr/lib
lrwxrwxrwx   1 root root          9 Feb 23 08:49 lib32 -> usr/lib32
lrwxrwxrwx   1 root root          9 Feb 23 08:49 lib64 -> usr/lib64
lrwxrwxrwx   1 root root         10 Feb 23 08:49 libx32 -> usr/libx32
drwx------   2 root root      16384 Mar  3 09:50 lost+found
drwxr-xr-x   2 root root       4096 Feb 23 08:50 media
drwxr-xr-x   2 root root       4096 Feb 23 08:50 mnt
drwxr-xr-x   2 root root       4096 Feb 23 08:50 opt
dr-xr-xr-x 265 root root          0 Mar 25 12:48 proc
drwx------   5 root root       4096 Mar 25 13:13 root
drwxr-xr-x  29 root root        860 Mar 25 12:50 run
lrwxrwxrwx   1 root root          8 Feb 23 08:49 sbin -> usr/sbin
drwxr-xr-x   6 root root       4096 Feb 23 08:57 snap
drwxr-xr-x   2 root root       4096 Feb 23 08:50 srv
-rw-------   1 root root 2068840448 Mar  3 09:51 swap.img
dr-xr-xr-x  13 root root          0 Mar 25 12:48 sys
drwxrwxrwt  12 root root       4096 Mar 25 13:02 tmp
drwxr-xr-x  14 root root       4096 Feb 23 08:53 usr
```

Rm -fr 命令：级联删除目录，包括目录下的文件也被删除；

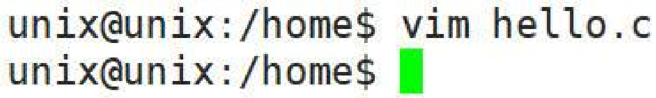


```
root@unix:/home/aaa# ls
bbb  ccc
root@unix:/home/aaa# cd ..
root@unix:/home# ls
aaa  unix
root@unix:/home# rm -fr aaa
root@unix:/home# ls
unix
root@unix:/home# 
```

3. vi 编辑器

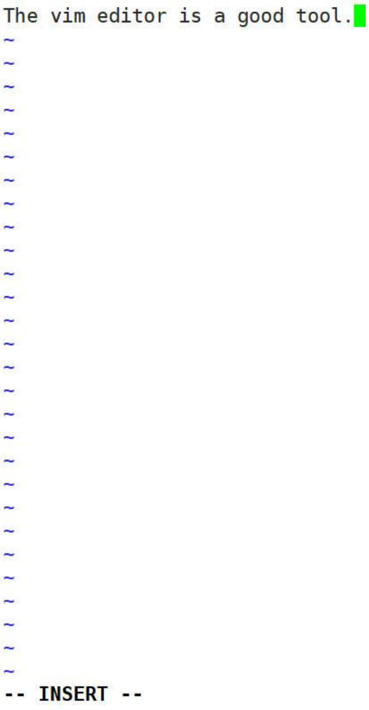
编辑器的编辑方式：

vi +文件名，初始进入文件为命令模式；



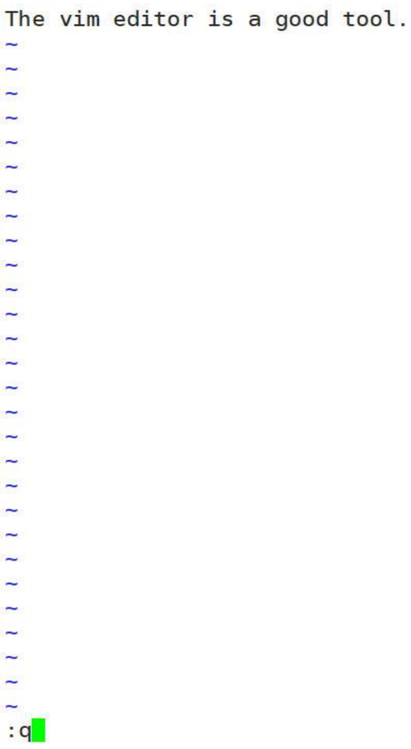
```
unix@unix:/home$ vim hello.c
unix@unix:/home$ 
```

文本输入模式；



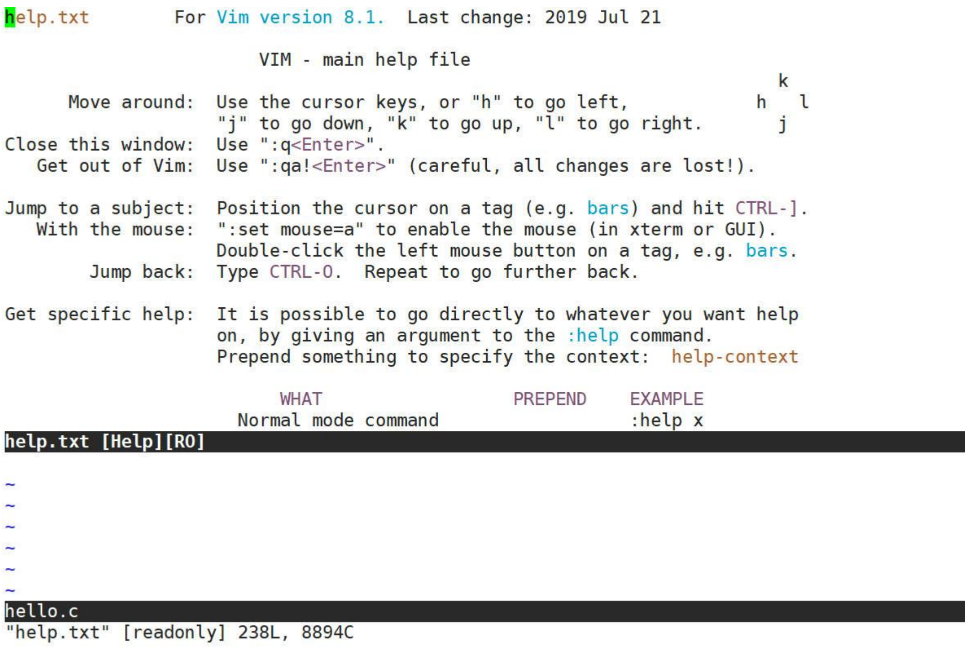
```
The vim editor is a good tool.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
-- INSERT --
```

退出编辑器：



```
The vim editor is a good tool.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:q
```

:help 命令：



```
help.txt        For Vim version 8.1.  Last change: 2019 Jul 21

                        VIM - main help file
                                                                     k
      Move around:  Use the cursor keys, or "h" to go left,        h   l
                    "j" to go down, "k" to go up, "l" to go right.   j
Close this window:  Use ":q<Enter>".
   Get out of Vim:  Use ":qa!<Enter>" (careful, all changes are lost!).

Jump to a subject:  Position the cursor on a tag (e.g. bars) and hit CTRL-].
   With the mouse:  ":set mouse=a" to enable the mouse (in xterm or GUI).
                    Double-click the left mouse button on a tag, e.g. bars.
        Jump back:  Type CTRL-O.  Repeat to go further back.

Get specific help:  It is possible to go directly to whatever you want help
                    on, by giving an argument to the :help command.
                    Prepend something to specify the context:  help-context

                          WHAT                  PREPEND    EXAMPLE
                      Normal mode command                  :help x
help.txt [Help][RO]
~
~
~
~
~
~
hello.c
"help.txt" [readonly] 238L, 8894C
```

移动光标：

H：光标向左移动一格；

J：光标向下移动一行

K：光标向上移动一行

L：光标向右移动一格

$：光标移动到当前行行尾；

0：光标移动到当前行行首；

W：光标右移一个字；

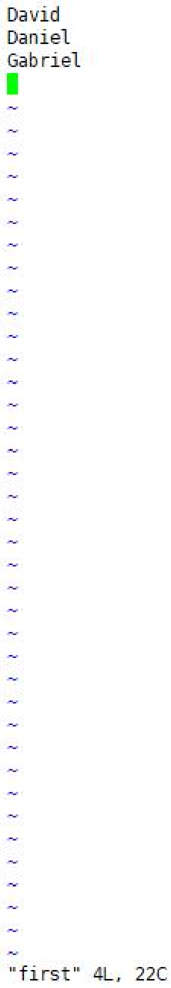
B：光标左移一个字；

E：光标移到字尾；

[Return]：光标移动到下一行行首，直至到达文件尾；

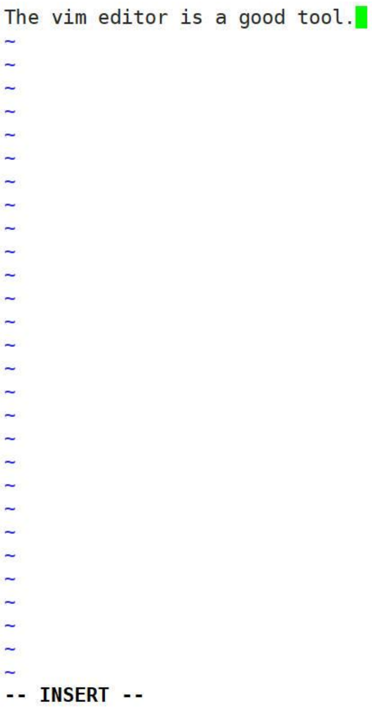
Vi 的模式：命令模式、输入模式、虚拟模式、命令行模式

命令模式：



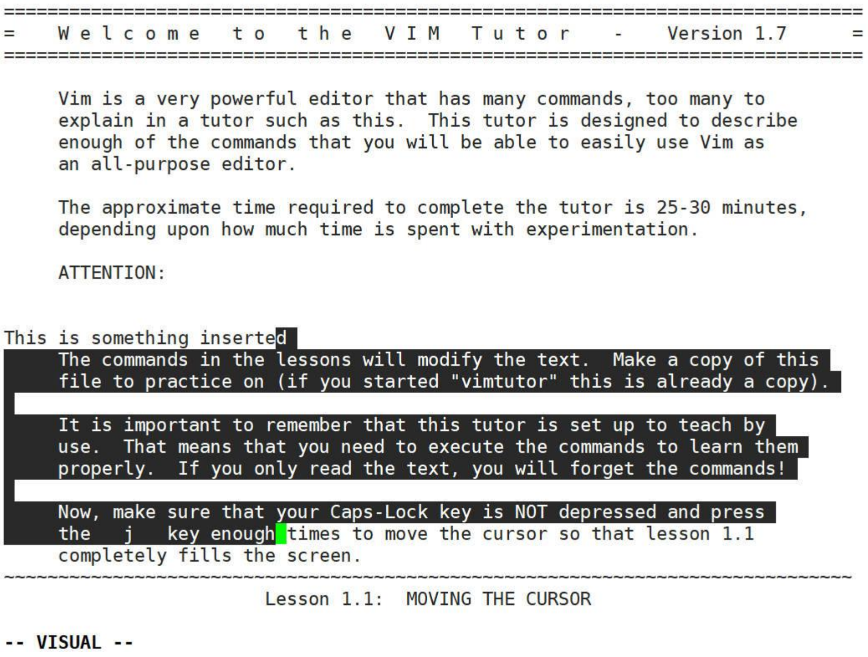
```
David
Daniel
Gabriel

~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"first" 4L, 22C
```

文本输入模式；



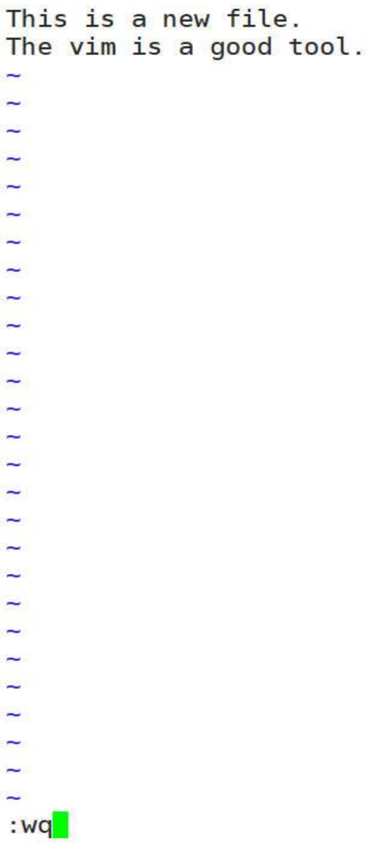
```
The vim editor is a good tool.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
-- INSERT --
```

虚拟模式：



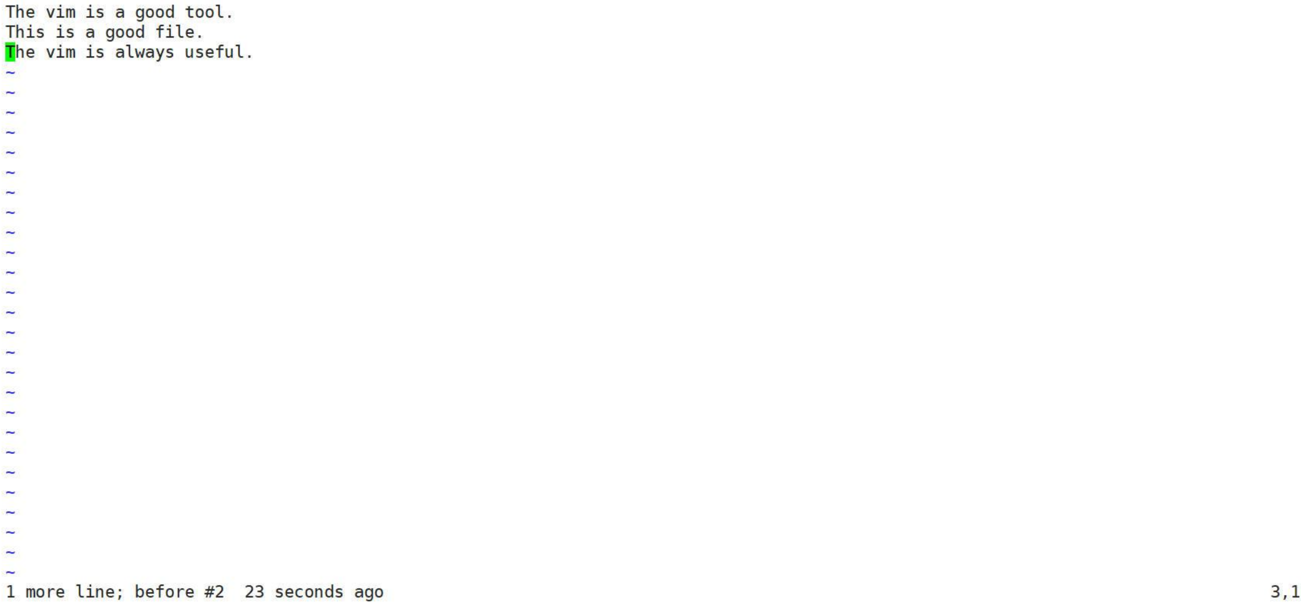
```
===============================================================================
=    W e l c o m e   t o   t h e   V I M   T u t o r    -    Version 1.7      =
===============================================================================

     Vim is a very powerful editor that has many commands, too many to
     explain in a tutor such as this.  This tutor is designed to describe
     enough of the commands that you will be able to easily use Vim as
     an all-purpose editor.

     The approximate time required to complete the tutor is 25-30 minutes,
     depending upon how much time is spent with experimentation.

     ATTENTION:


This is something inserted
     The commands in the lessons will modify the text.  Make a copy of this
     file to practice on (if you started "vimtutor" this is already a copy).

     It is important to remember that this tutor is set up to teach by
     use.  That means that you need to execute the commands to learn them
     properly.  If you only read the text, you will forget the commands!

     Now, make sure that your Caps-Lock key is NOT depressed and press
     the   j   key enough times to move the cursor so that lesson 1.1
     completely fills the screen.
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
                        Lesson 1.1:  MOVING THE CURSOR

-- VISUAL --
```

命令行模式：



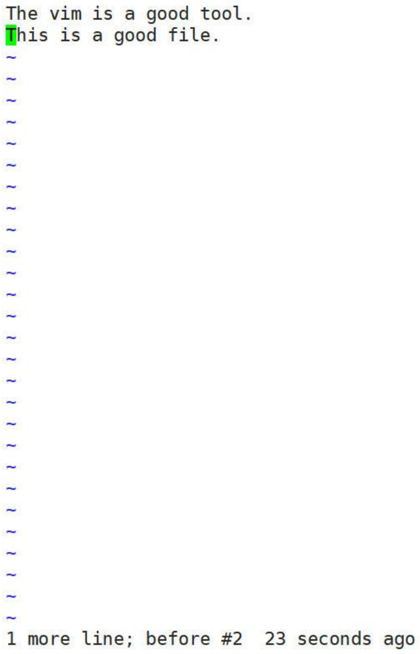
```
This is a new file.
The vim is a good tool.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:wq
```

4. 初始文件：



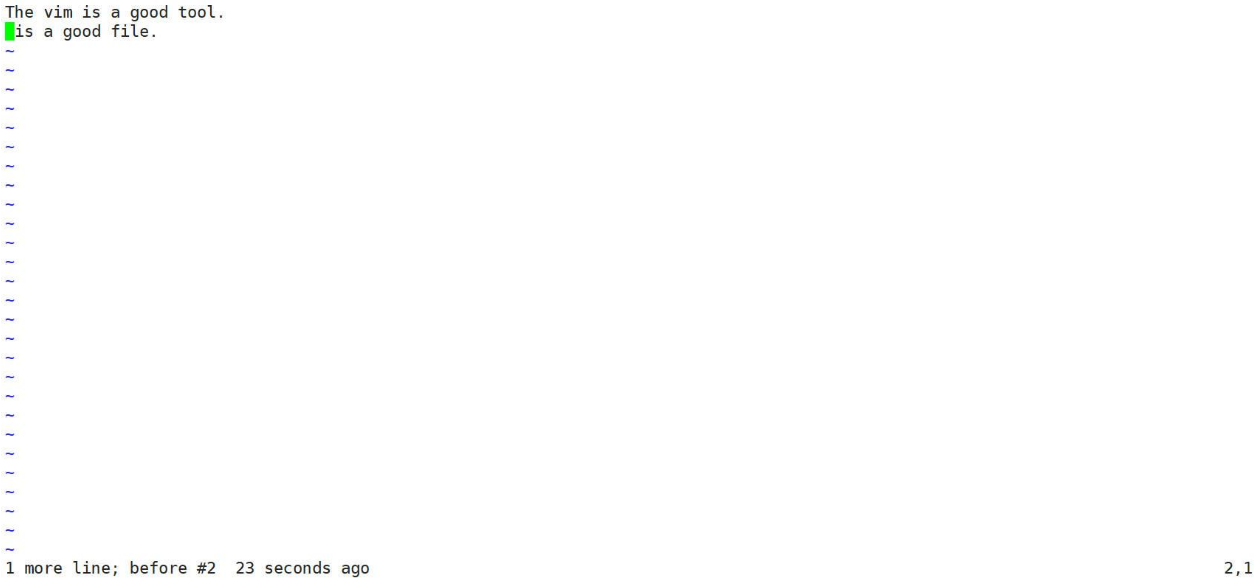
```
The vim is a good tool.
This is a good file.
The vim is always useful.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
1 more line; before #2  23 seconds ago                                                              3,1
```

a） dd 命令



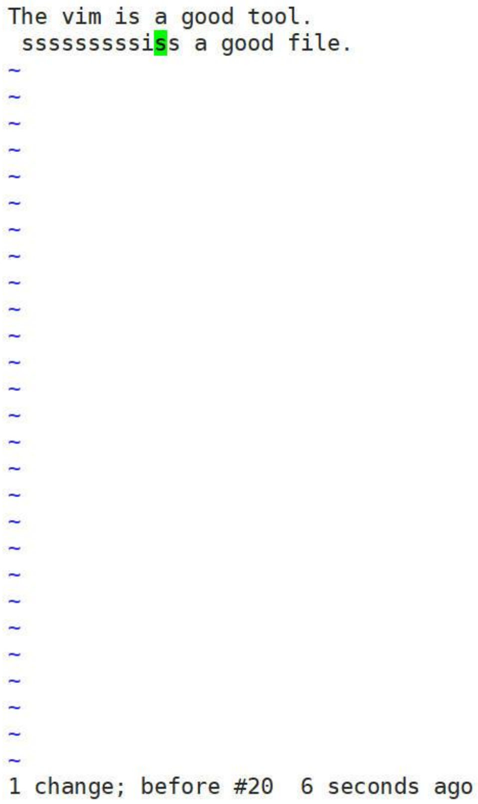
```
The vim is a good tool.
This is a good file.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
1 more line; before #2  23 seconds ago
```

X 命令：2x



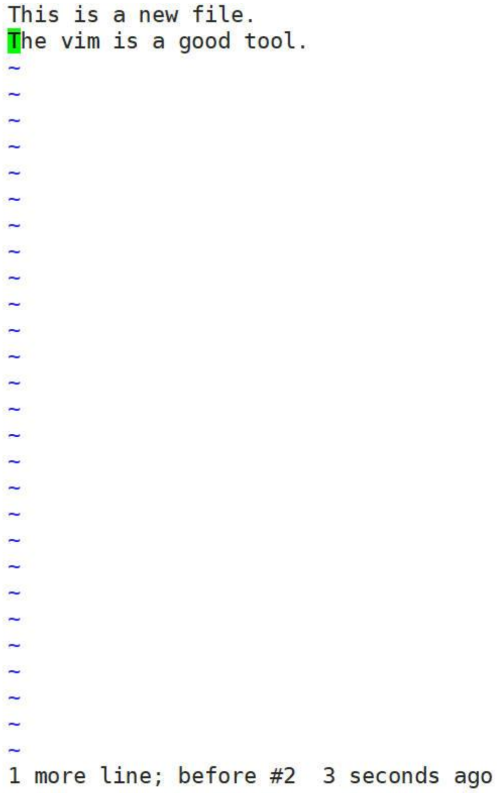
```
The vim is a good tool.
is a good file.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
1 more line; before #2  23 seconds ago                                                                      2,1
```

P 命令：



```
The vim is a good tool.
 sssssssssiss a good file.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
1 change; before #20  6 seconds ago
```

U 命令：

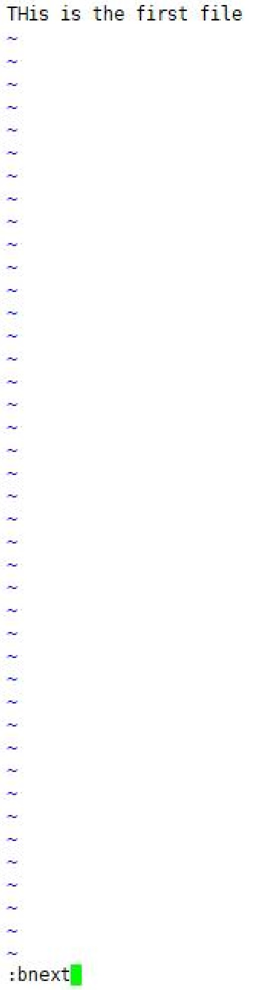


```
This is a new file.
The vim is a good tool.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
1 more line; before #2  3 seconds ago
```

b) 快速退出、切换 shell 的技巧(ZZ)

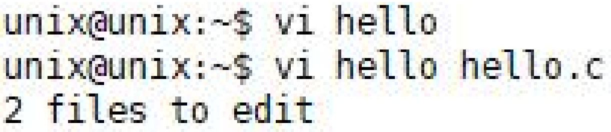
ZZ：快速退出 shell(超过 1 个文件不能使用)；

快速切换使用:bnext、:bprev；



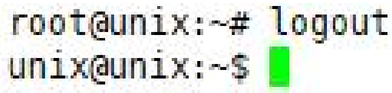
```
THis is the first file
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:bnext
```

```
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:bprev
```



```
unix@unix:~$ vi hello
unix@unix:~$ vi hello hello.c
2 files to edit
```

[Ctrl-d]快速退出登录：

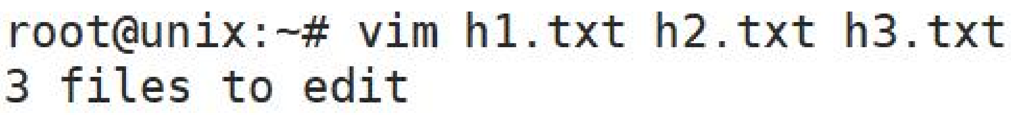


```
root@unix:~# logout
unix@unix:~$ 
```

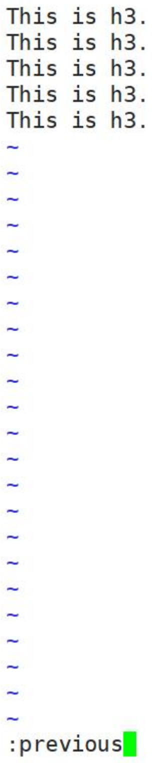
c） 编辑多个文件的技巧，利用 file、buffer、window 概念解释

通过 vi 编辑器同时打开多个文件，每个文件有一个 window，它们都在 buffers（缓冲区）中，使用:previous、:next 可以来回切换文件来进行编辑；

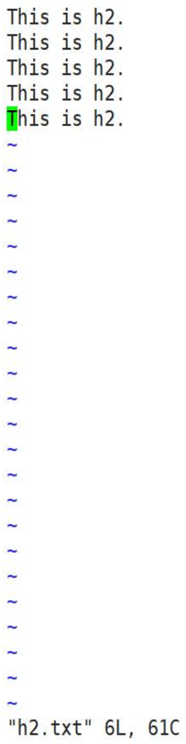
可以使用 tab，进行多窗口编辑，使用:tabnew 创建新窗口



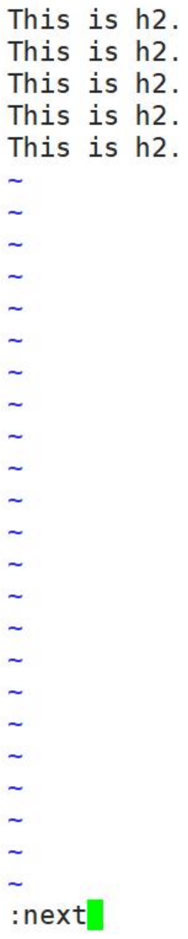
```
root@unix:~# vim h1.txt h2.txt h3.txt
3 files to edit
```



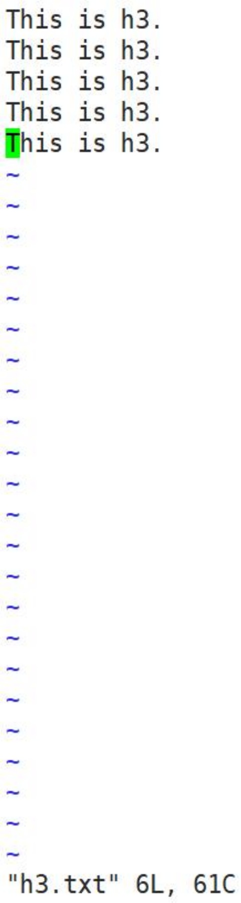
```
This is h3.
This is h3.
This is h3.
This is h3.
This is h3.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:previous
```



```
This is h2.
This is h2.
This is h2.
This is h2.
This is h2.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"h2.txt" 6L, 61C
```

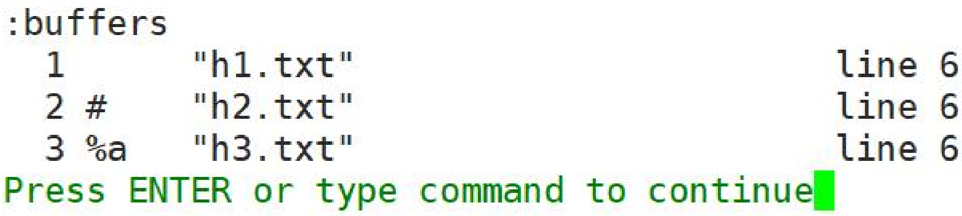


```
This is h2.
This is h2.
This is h2.
This is h2.
This is h2.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:next
```



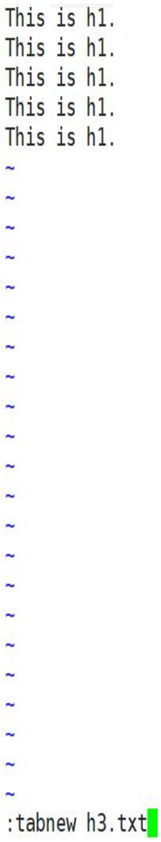
```
This is h3.
This is h3.
This is h3.
This is h3.
This is h3.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"h3.txt" 6L, 61C
```

Buffers 中状态：

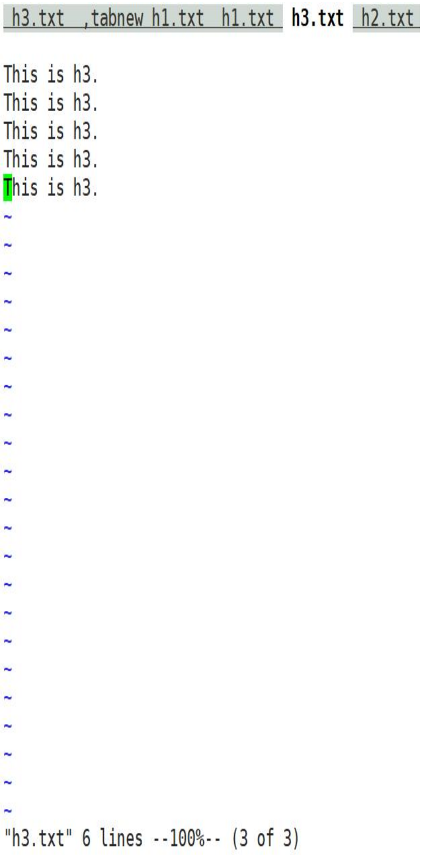


```
:buffers
  1      "h1.txt"                            line 6
  2 #    "h2.txt"                            line 6
  3 %a   "h3.txt"                            line 6
Press ENTER or type command to continue
```

Tab：创建新窗口编辑；

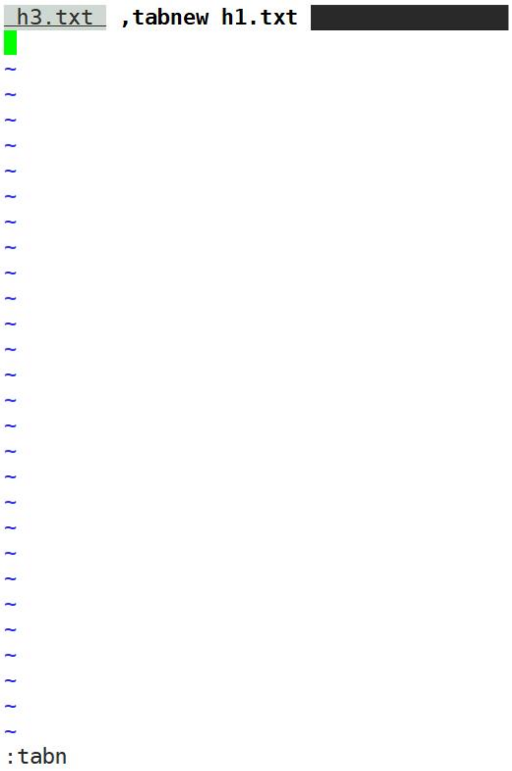


```
This is h1.
This is h1.
This is h1.
This is h1.
This is h1.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:tabnew h3.txt
```



```
 h3.txt  ,tabnew h1.txt  h1.txt  h3.txt  h2.txt

This is h3.
This is h3.
This is h3.
This is h3.
This is h3.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"h3.txt" 6 lines --100%-- (3 of 3)
```

切换窗口：



```
 h3.txt  ,tabnew h1.txt
 
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:tabn
```

d) 查看寄存器，解释有哪些寄存器

""：缓存最后一次操作内容；

"0 – "9：缓存最近操作内容，复制与删除有别；

"-：缓存行内删除内容；

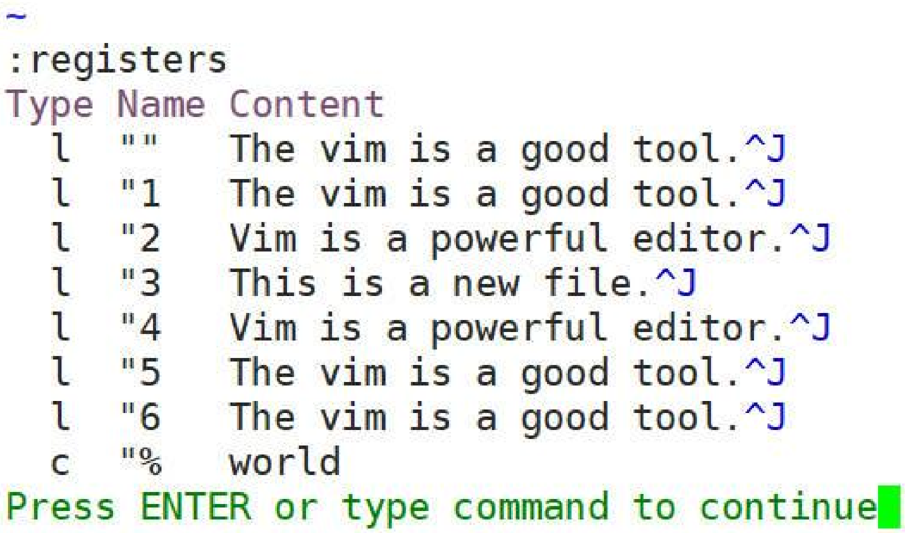
"a – "z 或"A – "Z：指定时可用；

":，".，"%，"#：分别缓存最近命令、最近插入文本、当前文件名、当前交替文件名；

"=：只读，用于执行表达式命令；

"_：不缓存操作内容（干净删除）；

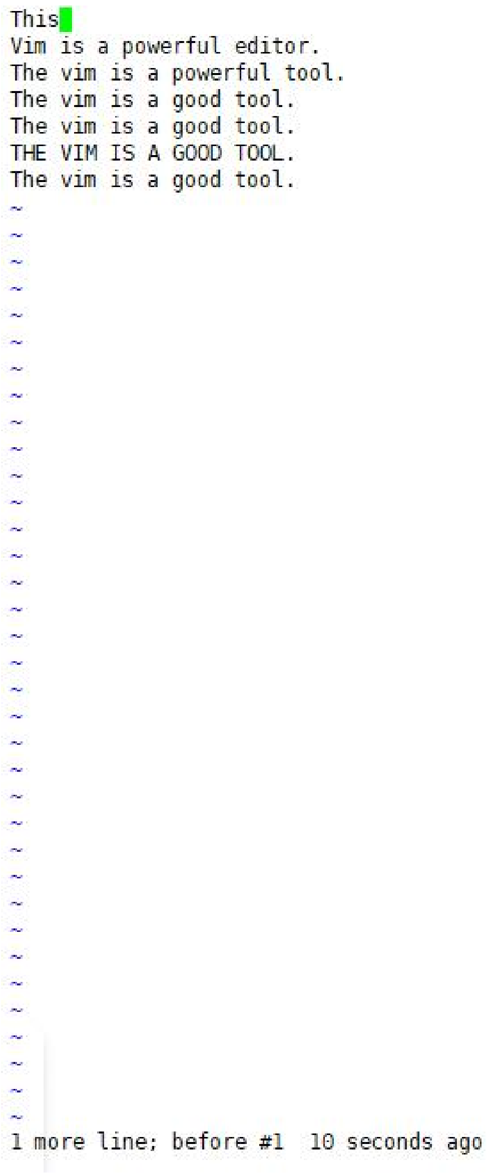
"：，缓存最近的搜索模式。



```
~
:registers
Type Name Content
  l  ""   The vim is a good tool.^J
  l  "1   The vim is a good tool.^J
  l  "2   Vim is a powerful editor.^J
  l  "3   This is a new file.^J
  l  "4   Vim is a powerful editor.^J
  l  "5   The vim is a good tool.^J
  l  "6   The vim is a good tool.^J
  c  "%   world
Press ENTER or type command to continue
```

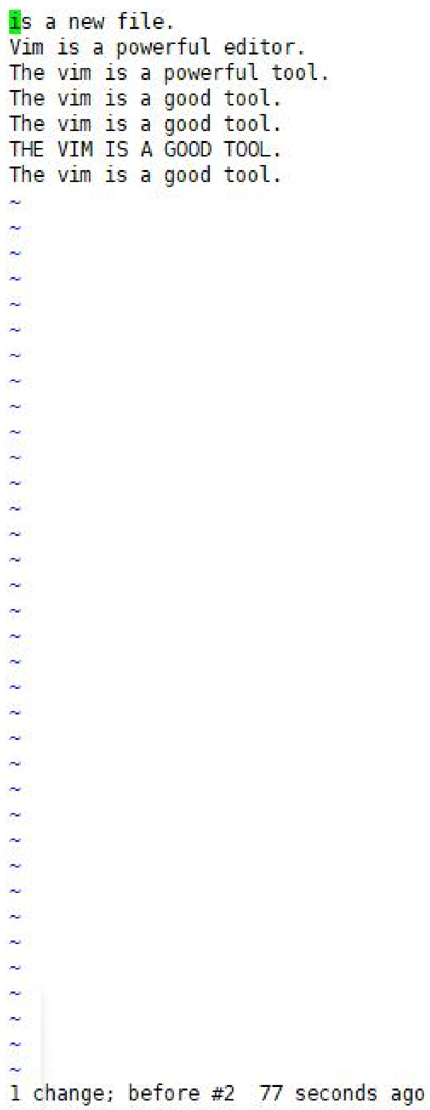
e） 观察 motion 与 range

d$：删除当前光标到行尾的文本：



```
This
Vim is a powerful editor.
The vim is a powerful tool.
The vim is a good tool.
The vim is a good tool.
THE VIM IS A GOOD TOOL.
The vim is a good tool.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
1 more line; before #1  10 seconds ago
```

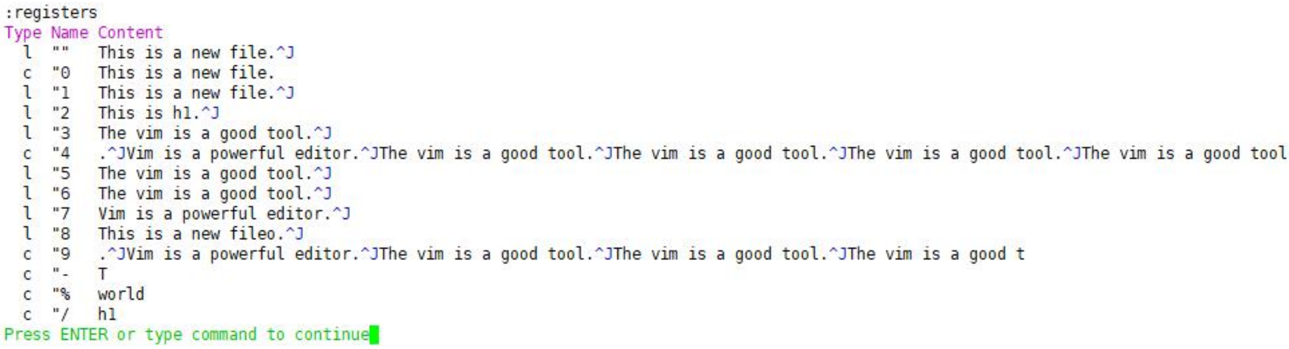
d0：删除当前光标到行首的文本：



```
is a new file.
Vim is a powerful editor.
The vim is a powerful tool.
The vim is a good tool.
The vim is a good tool.
THE VIM IS A GOOD TOOL.
The vim is a good tool.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
1 change; before #2  77 seconds ago
```

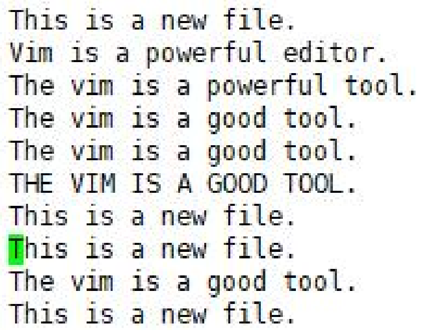
f） 寄存器组合命令，观察寄存器变化

“wdd：寄存器变化



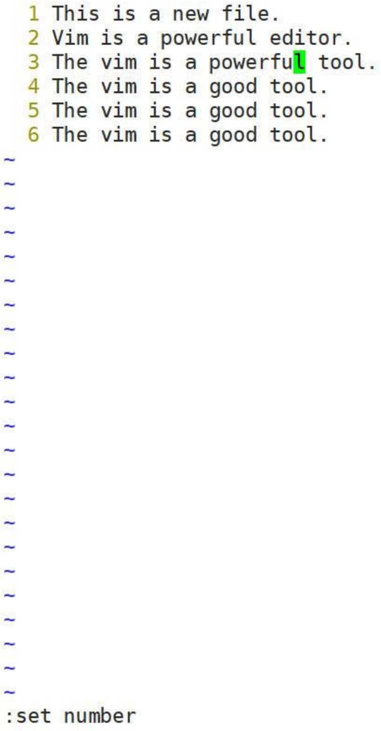
```
:registers
Type Name Content
  l  ""   This is a new file.^J
  c  "0   This is a new file.
  l  "1   This is a new file.^J
  l  "2   This is h1.^J
  l  "3   The vim is a good tool.^J
  c  "4   .^JVim is a powerful editor.^JThe vim is a good tool.^JThe vim is a good tool.^JThe vim is a good tool.^JThe vim is a good tool
  l  "5   The vim is a good tool.^J
  l  "6   The vim is a good tool.^J
  l  "7   Vim is a powerful editor.^J
  l  "8   This is a new fileo.^J
  c  "9   .^JVim is a powerful editor.^JThe vim is a good tool.^JThe vim is a good tool.^JThe vim is a good t
  c  "-   T
  c  "%   world
  c  "/   h1
Press ENTER or type command to continue
```

“wp：内容变化

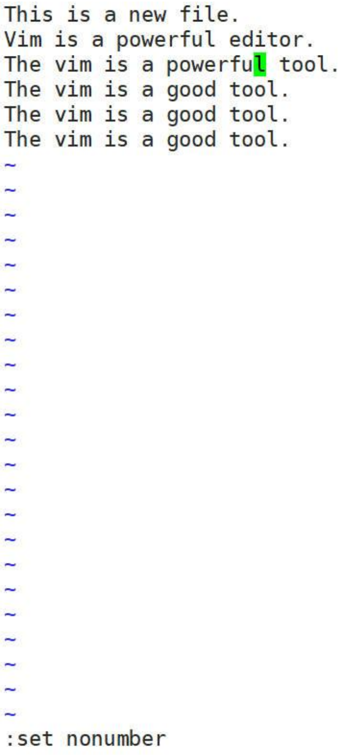


```
This is a new file.
Vim is a powerful editor.
The vim is a powerful tool.
The vim is a good tool.
The vim is a good tool.
THE VIM IS A GOOD TOOL.
This is a new file.
This is a new file.
The vim is a good tool.
This is a new file.
```

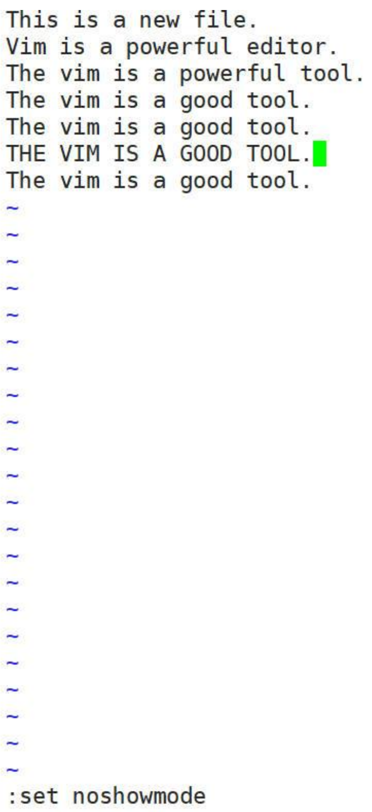
g） 选项，showmode、number



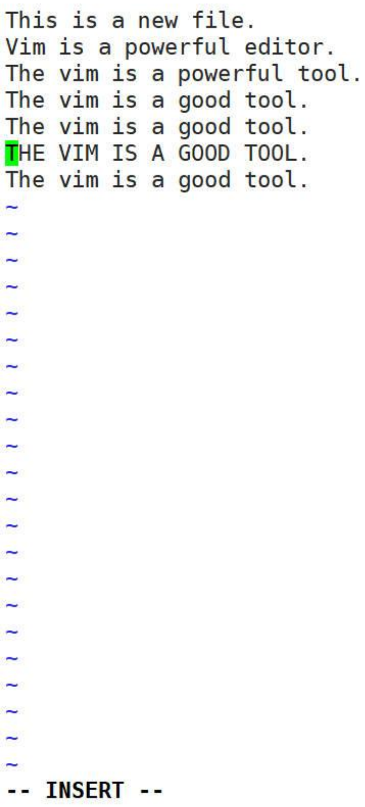
```
  1 This is a new file.
  2 Vim is a powerful editor.
  3 The vim is a powerful tool.
  4 The vim is a good tool.
  5 The vim is a good tool.
  6 The vim is a good tool.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:set number
```



```
This is a new file.
Vim is a powerful editor.
The vim is a powerful tool.
The vim is a good tool.
The vim is a good tool.
The vim is a good tool.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:set nonumber
```

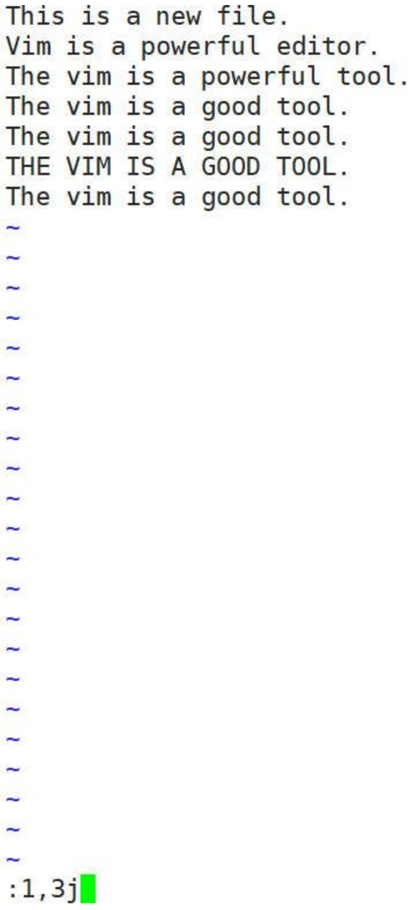


```
This is a new file.
Vim is a powerful editor.
The vim is a powerful tool.
The vim is a good tool.
The vim is a good tool.
THE VIM IS A GOOD TOOL.
The vim is a good tool.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:set noshowmode
```

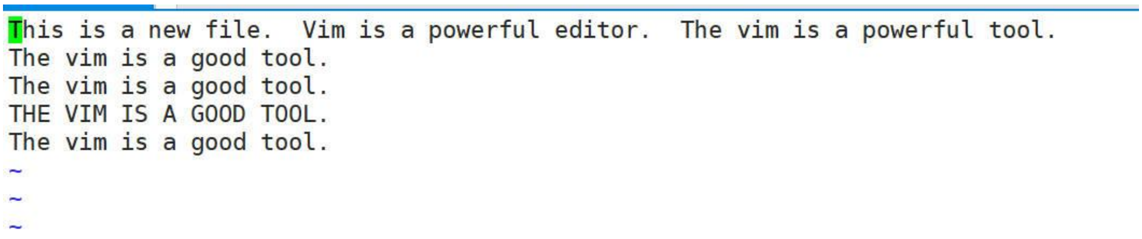


```
This is a new file.
Vim is a powerful editor.
The vim is a powerful tool.
The vim is a good tool.
The vim is a good tool.
THE VIM IS A GOOD TOOL.
The vim is a good tool.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
-- INSERT --
```

h） 连接两行 J



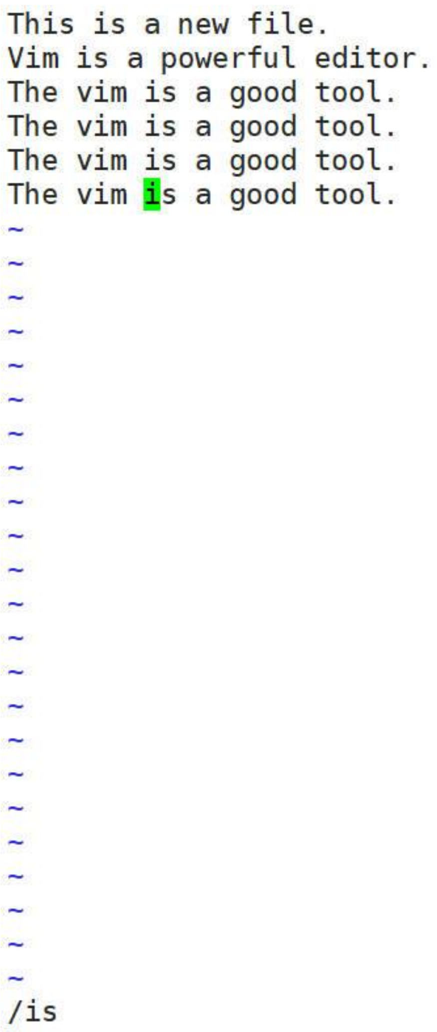
```
This is a new file.
Vim is a powerful editor.
The vim is a powerful tool.
The vim is a good tool.
The vim is a good tool.
THE VIM IS A GOOD TOOL.
The vim is a good tool.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:1,3j
```



```
This is a new file.  Vim is a powerful editor.  The vim is a powerful tool.
The vim is a good tool.
The vim is a good tool.
THE VIM IS A GOOD TOOL.
The vim is a good tool.
~
~
~
```

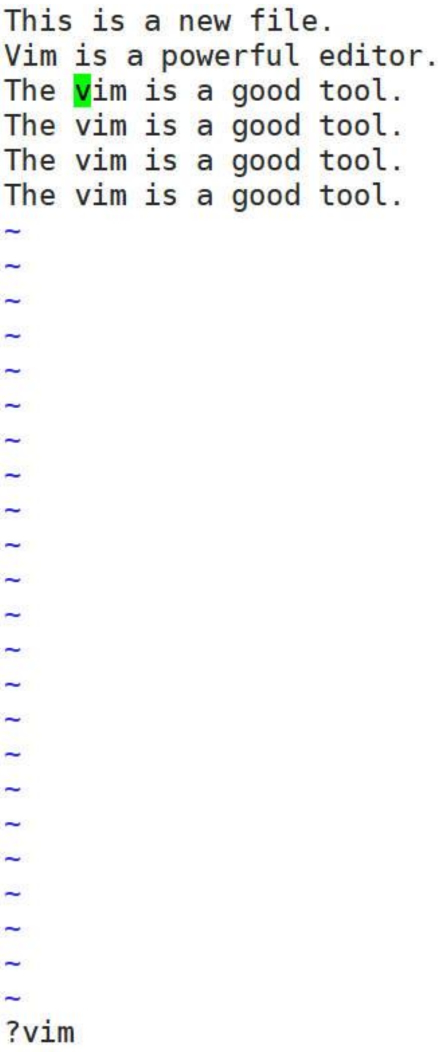
搜索字符串：

向前搜索：



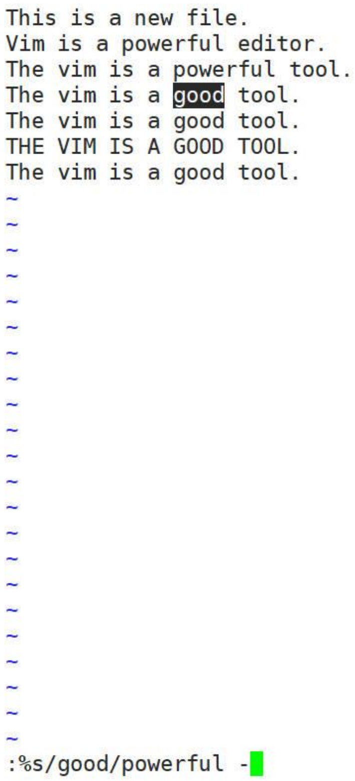
```
This is a new file.
Vim is a powerful editor.
The vim is a good tool.
The vim is a good tool.
The vim is a good tool.
The vim is a good tool.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
/is
```

向后搜索：

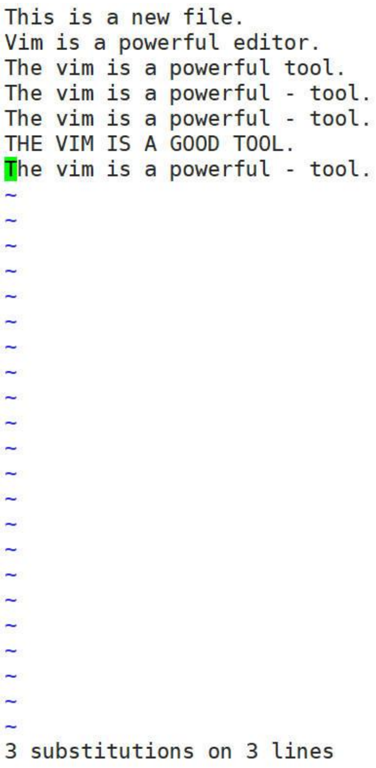


```
This is a new file.
Vim is a powerful editor.
The vim is a good tool.
The vim is a good tool.
The vim is a good tool.
The vim is a good tool.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
?vim
```

全局替换字符串：



```
This is a new file.
Vim is a powerful editor.
The vim is a powerful tool.
The vim is a good tool.
The vim is a good tool.
THE VIM IS A GOOD TOOL.
The vim is a good tool.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:%s/good/powerful -
```



```
This is a new file.
Vim is a powerful editor.
The vim is a powerful tool.
The vim is a powerful - tool.
The vim is a powerful - tool.
THE VIM IS A GOOD TOOL.
The vim is a powerful - tool.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
3 substitutions on 3 lines
```

十、实验结论：

这次实验对基本的一些系统操作命令和 vi 编辑器中的操作命令进行了实现和解释，让我对 UNIX 操作系统的基本命令有了更深的理解。

十一、总结及心得体会：

通过这次实验我了解了 UNIX 系统的基本命令，以及 vi 编辑器的基本操作；对常见的命令更加熟悉。

十二、对本实验过程及方法、手段的改进建议：

希望可以加深对命令的理解和对执行过程的了解。

报告评分：

指导教师签字：