

# Lab01 Assignment

班级: 192112  
学号: 19373073  
姓名: 何潇龙

## 实验准备

- 请安装一个合适的 Linux 系统, 你安装的 Linux 发行版及版本号是什么? 内核版本号是什么?

截图:

发行版本号:

```
charlot@charlot-virtual-machine:~$ cat /etc/issue
Ubuntu 16.04.6 LTS \n \l
```

内核版本号:

```
charlot@charlot-virtual-machine:~$ cat /proc/version
Linux version 4.15.0-45-generic (build@lcy01-amd64-025) (gcc version 5.4.0 20160609 (Ubuntu 5.4.0-6ubuntu1~16.04.10)) #48~16.04.1-Ubuntu SMP Tue Jan 29 18:03:19 UTC 2019
```

答案:

发行版: Ubuntu, 版本号: 16.04.6

内核版本号: 4.15.0-45-generic

- 查看你的根目录下有哪几个子目录, 每个子目录主要用来做什么用的?

截图:

```
charlot@charlot-virtual-machine:/$ ls
bin      dev      initrd.img  lost+found  opt      run      srv      usr
boot     etc      initrd.img.old  media      proc     sbin     sys      var
cdrom    home     lib         mnt         root     snap     tmp      vmlinuz
```

答案:

**/bin** 用来存放二进制可执行文件

**/dev** **device**, 任何设备都以文件的形式存放在这个目录中。

**/lost+found** 每一个分区在它的上级目录有该目录, 存放出现错误时保存的文件。

**/opt** **optional**, 可选的软件包, 即第三方软件。可以将除了系统自带软件之外的其他软件安装到这个目录下。

**/run** 最近一次开机后所产生的各项信息, 如当前的用户和正在运行中的守护进程等。

**/srv** **service**, 存放一些服务启动后所需的数据。

**/usr** 包含绝大部分所有用户(**users**)都能访问的应用程序和文件包含二进制文件, 库文件。文档和二级程序的源代码。

**/boot** 主要存放开机时用到的引导文件, 如**linux**内核文件和开机菜单与开机所需要的配置文件。

**/etc** 配置文件、启动脚本等 (**etc**)包含所有程序所需的配置文件以及系统的配置文件。也包含了用于启动/停止单个程序的启动和关闭**shell**脚本。

**/media** **media** (媒体), 存放可移除的媒体设备

**/proc** 特殊的动态目录, 用以维护系统信息和状态, 包括当前运行中进程 (**processes**) 信息。包含系统进程的相关信息, 是一个虚拟的文件系统, 包含有关正在运行的进程的信息, 系统资源以文本信息形式存在。

**/sbin** 存放一些只有**root**账户才有权限执行的可执行文件

**/sys** **system**, 与**proc**类似也是一个虚拟文件系统, 存放系统核心与硬件相关信息管理设备文件。

**/var** 经常变化的(**variable**)文件, 诸如日志或数据库等代表变量文件。

**/cdrom** 专门用来挂载光盘的目录, 有些发行版将该目录放在**media**或**mnt**目录下。

`/home` 系统默认的用户的家目录，每当新建一个用户系统都会在这个目录下创建以该用户名为名称的目录作为该用户的家目录。并且在命令行中代表当前用户的家目录，`redhat`表示

`/lib` `library`，存放着系统开机时所需的函数库以及`/bin`和`/sbin`目录下的命令会调用的函数库。

`/mnt` `mount`，临时挂载的设备文件，临时安装目录，系统管理员可以挂载文件系统。是系统管理员临时安装文件的系统安装点。

`/root` 系统管理员`root`的主目录。

`/snap`：这是使用`snap`安装软件产生的目录

`/tmp` `temporary`，存放系统运行过程中使用的一些临时文件，可以被所有用户访问，系统重启时会清空该目录。

- 查看自己的 ip 地址，并 ping 一下 baidu.com 看网络是否连通？

命令：

查看ip地址的命令：`hostname -I`

看网络是否连通的命令：`ping baidu.com`

截图：

```
charlot@charlot-virtual-machine:~$ hostname -I
192.168.33.130

charlot@charlot-virtual-machine:~$ ping baidu.com
PING baidu.com (39.156.69.79) 56(84) bytes of data.
64 bytes from 39.156.69.79: icmp_seq=1 ttl=128 time=8.22 ms
64 bytes from 39.156.69.79: icmp_seq=2 ttl=128 time=7.64 ms
64 bytes from 39.156.69.79: icmp_seq=3 ttl=128 time=7.42 ms
64 bytes from 39.156.69.79: icmp_seq=4 ttl=128 time=7.99 ms
64 bytes from 39.156.69.79: icmp_seq=5 ttl=128 time=8.09 ms
64 bytes from 39.156.69.79: icmp_seq=6 ttl=128 time=7.81 ms
64 bytes from 39.156.69.79: icmp_seq=7 ttl=128 time=7.95 ms
64 bytes from 39.156.69.79: icmp_seq=8 ttl=128 time=37.8 ms
64 bytes from 39.156.69.79: icmp_seq=9 ttl=128 time=8.13 ms
64 bytes from 39.156.69.79: icmp_seq=10 ttl=128 time=7.97 ms
```

- 用软件安装命令下载 build-essential。

命令：`sudo apt install build-essential`

截图：

```
charlot@charlot-virtual-machine:~$ sudo apt install build-essential
[sudo] password for charlot:
Reading package lists... Done
Building dependency tree
Reading state information... Done
build-essential is already the newest version (12.1ubuntu2).
0 upgraded, 0 newly installed, 0 to remove and 426 not upgraded.
```

## 1. Linux 命令操作

- Linux 命令行操作，请用你学到的 Linux 命令，实现以下操作，给出每一步你的命令行截图。
  - 用 `wget` 从 `https://github.com/BUAA-SE-2021/patpat/releases/download/v0.1.3/patpat-linux-amd64` 下载你们的 Linux 版 OOP 课 Java 自助评测机 `patpat`
  - `https://github.com/BUAA-SE-2021/sp-1abs/lab01/lab01.zip` 处下载实验压缩包
  - 解压 `lab01.zip`。

- 进入 lab01 目录,进入子目录 etc , 打印当前路径并在当前路径下创建名为 a1 的目录, 并在 a1 目录中创建名为 b1 的目录。
- 进入 b1 目录中, 创建两个文件 a.txt , b.txt 。
- 退回子目录 etc , 删除目录 a1 。
- 将 etc 目录下所有以 tmp 开头的文件移动到 lab01 目录下的 Download 目录下的 tmp 目录中。
- 查看 tmp 目录下 a1005.cpp 的内容。
- 查看 tmp 目录下 a1009.cpp 的前十行和后十行。
- 将 tmp 目录下的所有文件打成一个 tar 包, 并命名为并命名为 tmp.tar.gz 。
- 返回 lab01 目录, 列出当前目录下的文件大小。
- 用命令找出空目录并将空目录删除。

命令:

```
# 不完整的部分自行补充, 填写必要注释
code wget https://github.com/BUAA-SE-2021/patpat/releases/download/v0.1.3/patpat-linux-amd64# 下载
code wget https://hub.fastgit.org/BUAA-SE-2021/sp-labs/raw/main/lab01/lab01.zip# 下载实验资料, 这里因为连不上github所以采用了另一个链接下载
code unzip lab01.zip# 解压
code cd lab01/etc# 进入目录
code pwd# 打印当前路径
code mkdir a1# 创建目录`a1`
code mkdir a1/b1# 创建子目录`b1`
code cd ../..# 返回子目录`etc`
code rm -r a1# 删除`a1`
code find . -name 'tmp*' -exec mv {} ../Download/tmp \;;# 移动
code cat ../Download/tmp/a1005.cpp# 查看
code head -10 ../Download/tmp/a1009.cpp tail -10 ../Download/tmp/a1009.cpp# 查看前十行和后十行
code tar -cf tmp.tar.gz tmp# 打包, 首先要进到Download目录里, cd .. cd Download
code cd ../..# 返回`lab01`目录
code du -h# 列出当前目录下的文件大小
code find -type d -empty# 用命令找出空目录
code rm -r ./upload# 删除空目录
```

必要的实验截图(如查看前十行、查看文件大小的展示效果)

1、下载patpat

```

charlot@charlot-virtual-machine:~$ wget http://github.com/BUAA-SE-2021/patpat/releases/download/v0.1.3/patpat-linux-amd64
--2021-03-21 21:00:31-- http://github.com/BUAA-SE-2021/patpat/releases/download/v0.1.3/patpat-linux-amd64
Resolving github.com (github.com)... 13.229.188.59
Connecting to github.com (github.com)|13.229.188.59|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://github.com/BUAA-SE-2021/patpat/releases/download/v0.1.3/patpat-linux-amd64 [following]
--2021-03-21 21:00:31-- https://github.com/BUAA-SE-2021/patpat/releases/download/v0.1.3/patpat-linux-amd64
Connecting to github.com (github.com)|13.229.188.59|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://github-releases.githubusercontent.com/336680845/033f0900-7b8d-11eb-8a5f-e0515ba1ffab?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWNJYAX4CSVEH53A%2F20210321%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20210321T130032Z&X-Amz-Expires=300&X-Amz-Signature=b9b7539ef53526c9bc0bbd834ffebbbadc7b3a43480e62d5a3b6f32d08cab73f&X-Amz-SignedHeaders=host&actor_id=0&key_id=0&repo_id=336680845&response-content-disposition=attachment%3B%20filename%3Dpatpat-linux-amd64&response-content-type=application%2Foctet-stream [following]
--2021-03-21 21:00:32-- https://github-releases.githubusercontent.com/336680845/033f0900-7b8d-11eb-8a5f-e0515ba1ffab?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWNJYAX4CSVEH53A%2F20210321%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20210321T130032Z&X-Amz-Expires=300&X-Amz-Signature=b9b7539ef53526c9bc0bbd834ffebbbadc7b3a43480e62d5a3b6f32d08cab73f&X-Amz-SignedHeaders=host&actor_id=0&key_id=0&repo_id=336680845&response-content-disposition=attachment%3B%20filename%3Dpatpat-linux-amd64&response-content-type=application%2Foctet-stream
Resolving github-releases.githubusercontent.com (github-releases.githubusercontent.com)... 185.199.110.154, 185.199.111.154, 185.199.108.154, ...
Connecting to github-releases.githubusercontent.com (github-releases.githubusercontent.com)|185.199.110.154|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 6299648 (6.0M) [application/octet-stream]
Saving to: 'patpat-linux-amd64'

patpat-linux-amd64 100%[=====>] 6.01M 5.41MB/s in 1.1s

```

## 2、下载实验材料

```

charlot@charlot-virtual-machine:~$ wget http://hub.fastgit.org/BUAA-SE-2021/sp-labs/raw/main/lab01/lab01.zip
--2021-03-21 21:05:11-- http://hub.fastgit.org/BUAA-SE-2021/sp-labs/raw/main/lab01/lab01.zip
Resolving hub.fastgit.org (hub.fastgit.org)... 89.31.125.6
Connecting to hub.fastgit.org (hub.fastgit.org)|89.31.125.6|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://hub.fastgit.org/BUAA-SE-2021/sp-labs/raw/main/lab01/lab01.zip [following]
--2021-03-21 21:05:11-- https://hub.fastgit.org/BUAA-SE-2021/sp-labs/raw/main/lab01/lab01.zip
Connecting to hub.fastgit.org (hub.fastgit.org)|89.31.125.6|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://raw.fastgit.org/BUAA-SE-2021/sp-labs/main/lab01/lab01.zip [following]
--2021-03-21 21:05:12-- https://raw.fastgit.org/BUAA-SE-2021/sp-labs/main/lab01/lab01.zip
Resolving raw.fastgit.org (raw.fastgit.org)... 104.149.187.202
Connecting to raw.fastgit.org (raw.fastgit.org)|104.149.187.202|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2318 (2.3K) [application/zip]
Saving to: 'lab01.zip'

lab01.zip          100%[=====>] 2.26K --.-KB/s in 0s

2021-03-21 21:05:13 (497 MB/s) - 'lab01.zip' saved [2318/2318]

```

## 3、解压lab01

```
charlot@charlot-virtual-machine:~$ unzip lab01.zip
Archive:  lab01.zip
  creating: lab01/
  creating: lab01/Download/
  creating: lab01/Download/tmp/
  inflating: lab01/Download/tmp/a1005.cpp
  inflating: lab01/Download/tmp/a1009.cpp
  creating: lab01/etc/
  creating: lab01/etc/c1/
  extracting: lab01/etc/c1/sc.txt
  extracting: lab01/etc/c1/tmp1.c
  extracting: lab01/etc/c1/tmp2.cpp
  extracting: lab01/etc/c1/tmptmp.txt
  creating: lab01/upload/
```

#### 4、进入目录

```
charlot@charlot-virtual-machine:~$ cd lab01/etc
charlot@charlot-virtual-machine:~/lab01/etc$
```

#### 5、打印路径

```
charlot@charlot-virtual-machine:~/lab01/etc$ pwd
/home/charlot/lab01/etc
```

#### 6、创建目录a1

```
charlot@charlot-virtual-machine:~/lab01/etc$ mkdir a1
```

#### 7、创建子目录b1

```
charlot@charlot-virtual-machine:~/lab01/etc$ mkdir a1/b1
```

在b1中创建两个文件

```
charlot@charlot-virtual-machine:~/lab01/etc$ cd a1/b1
charlot@charlot-virtual-machine:~/lab01/etc/a1/b1$ touch a.txt
charlot@charlot-virtual-machine:~/lab01/etc/a1/b1$ touch b.txt
charlot@charlot-virtual-machine:~/lab01/etc/a1/b1$
```

#### 8、返回子目录etc

```
charlot@charlot-virtual-machine:~/lab01/etc/a1/b1$ cd ../../
charlot@charlot-virtual-machine:~/lab01/etc$
```

#### 9、删除a1

```
charlot@charlot-virtual-machine:~/lab01/etc$ rm -r a1
```

#### 10、移动

```
charlot@charlot-virtual-machine:~/lab01/etc$ find . -name 'tmp*' -exec mv {} ../
Download/tmp/ \;
```

#### 11、查看



```

charlot@charlot-virtual-machine:~/lab01/etc$ cat ../Download/tmp/a1005.cpp
#include <iostream>
#include <cstdio>
#include <algorithm>

int hash[10500];
void add(int num){
    if(num%2==0){
        num/=2;
    }
    else num=(num*3+1)/2;
    while (num!=1)
    {
        if(hash[num]==0){
            hash[num]=1;
        }
        if(num%2==0){
            num/=2;
        }
        else {
            num=num*3+1;
            num/=2;
        }
    }
}

```

12、查看前十行和后十行

```

charlot@charlot-virtual-machine:~/lab01/etc$ head -10 ../Download/tmp/a1009.cpp
#include <iostream>
#include <cstdio>

using namespace std;

char str[100][80];
int main()
{
    int i=0;
    while (~scanf("%s",str+i))
charlot@charlot-virtual-machine:~/lab01/etc$ tail -10 ../Download/tmp/a1009.cpp
{
    i++;
}
for(int j=i-1;j>=0;j--){
    if(j==i-1)
        printf("%s",str+j);
    else printf(" %s",str+j);
}
return 0;
}

```

13、打包

```

charlot@charlot-virtual-machine:~/lab01/etc$ cd ..
charlot@charlot-virtual-machine:~/lab01$ cd Download
charlot@charlot-virtual-machine:~/lab01/Download$ tar -cf tmp.tar.gz tmp

```

14、返回lab01目录

```

charlot@charlot-virtual-machine:~/lab01/Download$ cd ..
charlot@charlot-virtual-machine:~/lab01$

```

15、列出当前目录下的文件大小

```

charlot@charlot-virtual-machine:~/lab01$ du -h
12K      ./Download/tmp
28K      ./Download
4.0K     ./etc/c1
8.0K     ./etc
4.0K     ./upload
44K      .

```

16、用命令找出空目录

```
charlot@charlot-virtual-machine:~/lab01$ find -type d -empty
./upload
```

17、删除空目录

```
charlot@charlot-virtual-machine:~/lab01$ rm -r ./upload
```

## 2. vi 模式

- vi 编辑器有哪几种模式？简述这几种模式间如何互相切换？

模式：

- 1、命令模式(command mode)
- 2、插入模式(insert mode)
- 3、底行模式(last line mode)(ppt上是末行模式)

如何切换：

命令模式：使用 vi 打开文件后默认进入该模式

插入模式：在命令模式下输入操作符(如 i)进入插入模式，ESC 从编辑模式返回命令模式

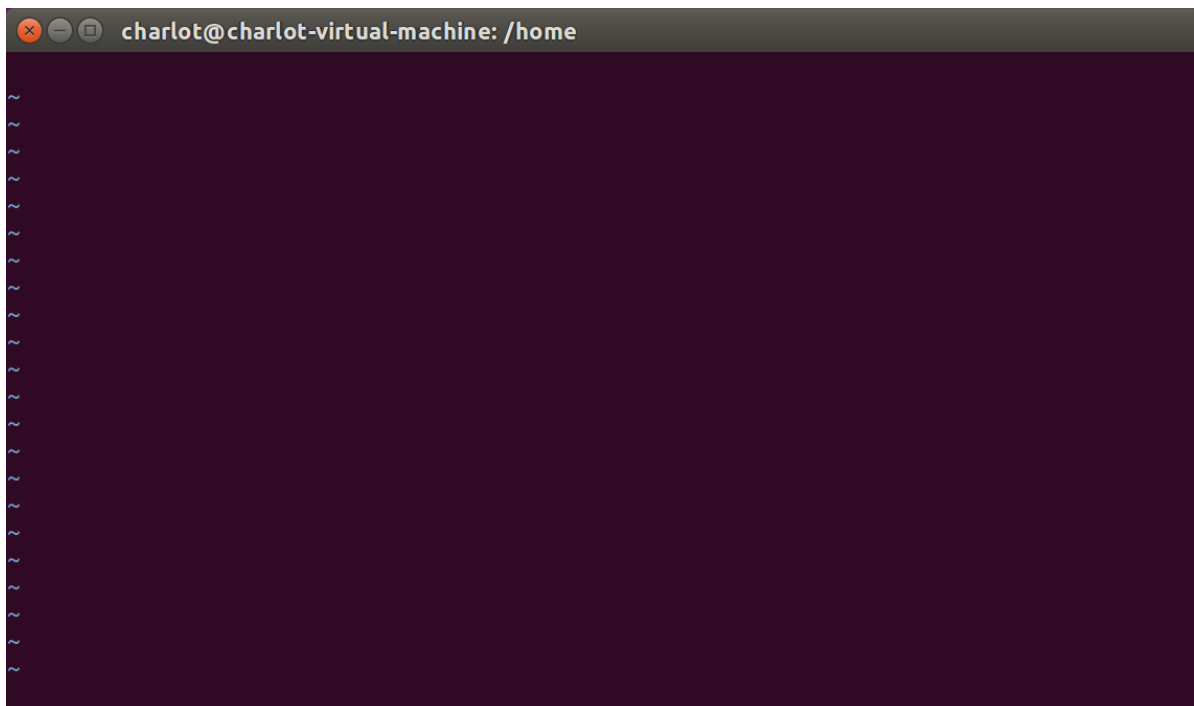
底行模式：在命令模式下输入 : 或 / 进入底行模式，ESC 清空底行或返回命令模式

## 3. vi 命令

写出以下一系列操作使用的命令（底行模式的命令加上 : 或 / ）：

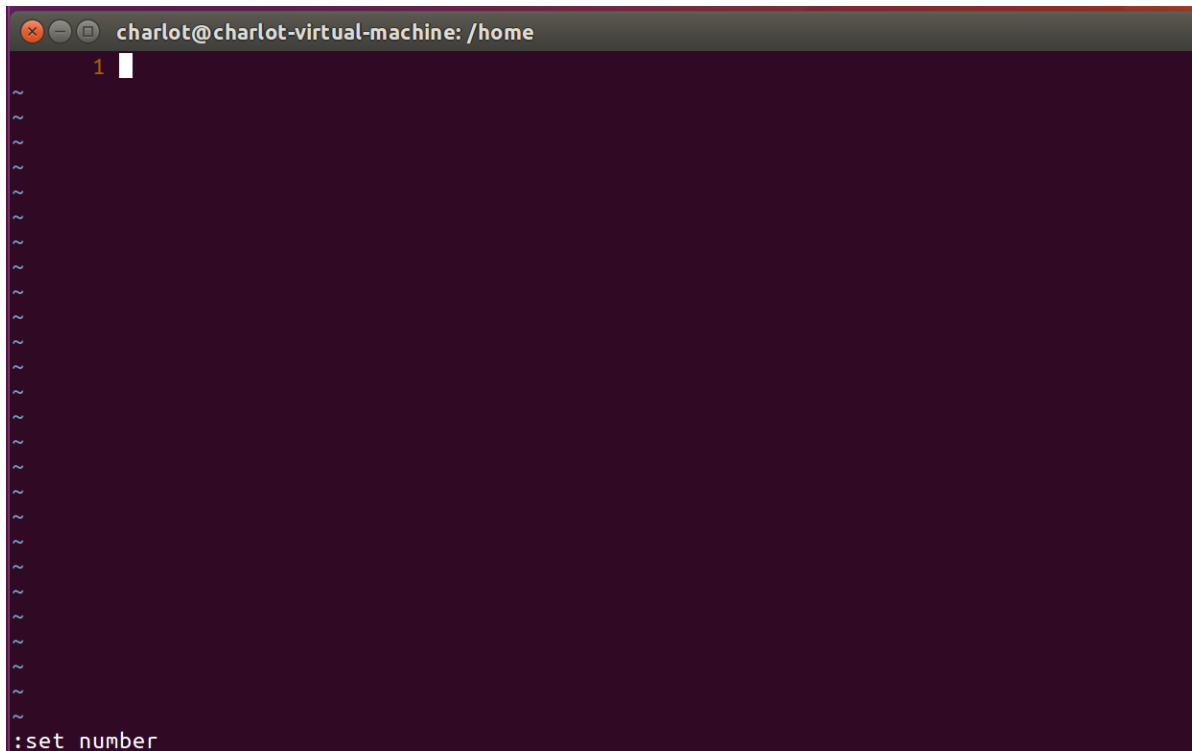
### 3.1. 用 vi 在当前用户家目录下新建文件 testfile 并打开

```
vi testfile
```



## 3.2. 设置显示行号

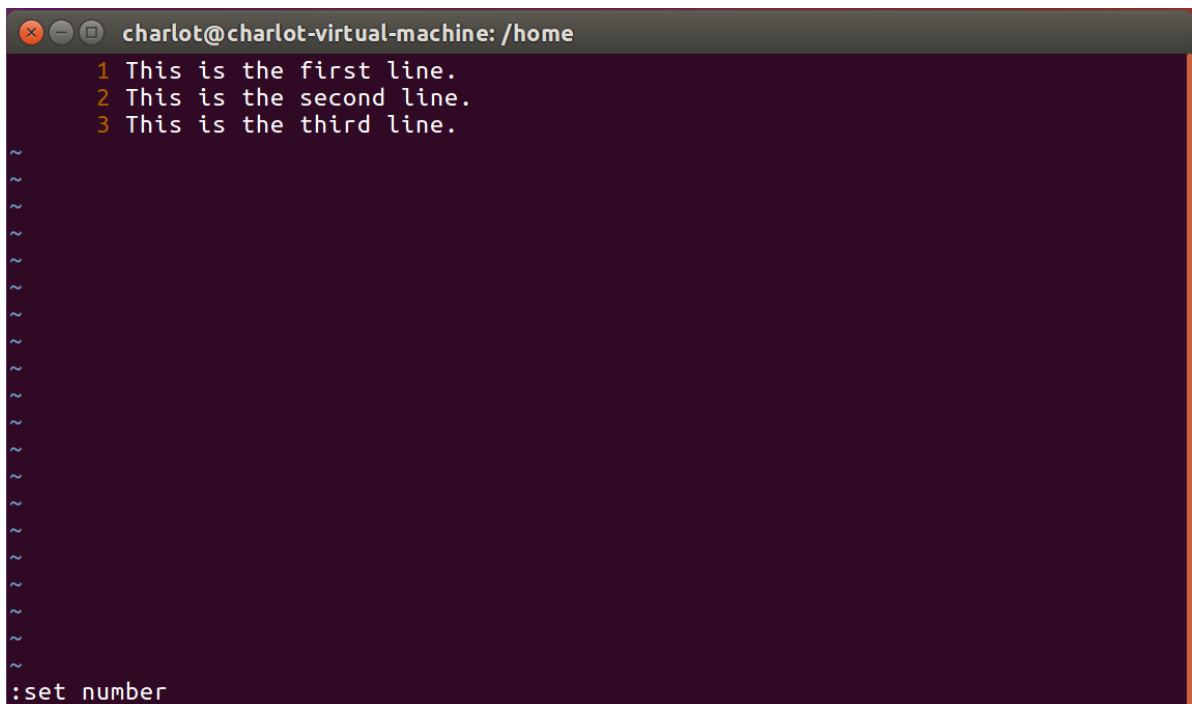
```
:set number
```



A screenshot of a terminal window titled "charlot@charlot-virtual-machine: /home". The terminal shows a Vim editor with line numbers displayed on the left margin. The first line is numbered "1" and contains a cursor. The rest of the lines are numbered with tilde symbols (~). At the bottom of the terminal, the command ":set number" is entered.

## 3.3. 进入 insert mode , 输入 3 行文本

```
This is the first line.  
This is the second line.  
This is the third line.
```



A screenshot of a terminal window titled "charlot@charlot-virtual-machine: /home". The terminal shows a Vim editor with three lines of text entered in insert mode. The lines are numbered 1, 2, and 3 on the left margin. The text of the lines is "This is the first line.", "This is the second line.", and "This is the third line.". The rest of the lines are numbered with tilde symbols (~). At the bottom of the terminal, the command ":set number" is entered.









这次试验让我对linux有了初步的了解，稍微熟悉了一下各种指令和操作。但是感觉命令实在还是有点记不下来，还需要多加练习变得更加熟练。然后感觉虚拟机上的操作还是挺有意思的，vi编辑器那一块一开始因为不熟悉各种模式疯狂出错，后面慢慢习惯了之后还是有点成就感的。总之要多练！然后再也不要压ddl了！！早布置早完成！