

# 数据科学导论实验报告

## 实验一 常用的 Linux 操作和 Hadoop 操作



学 生：

年 级：2021 级

专 业：软件工程

重庆大学大数据与软件学院

2023 年 10 月 12 日

## 一、 实验目的：

实验一的目的是让学生掌握常用的 Linux 操作和 Hadoop 操作。这个实验旨在培养学生或参与者在 Linux 环境下进行基本的文件和系统管理，同时了解如何使用 Hadoop 框架进行分布式数据处理，为后续上机实验做准备。

## 二、 实验要求

### (一) 熟悉常用的 Linux 操作

1. 使用`cd`命令：
  - (1) 切换到目录`/usr/local`
  - (2) 返回上一级目录
  - (3) 返回用户的主文件夹
  
2. 使用`ls`命令：
  - (4) 查看目录`/usr`下的所有文件
  
3. 使用`mkdir`命令：
  - (5) 进入`/tmp`目录，创建一个名为`a`的目录，并查看存在多少目录
  - (6) 创建目录`a1/a2/a3/a4`
  
4. 使用`rmdir`命令：
  - (7) 删除在上例中创建的目录`a`（位于`/tmp`下）
  - (8) 删除目录`a1/a2/a3/a4`，并查看存在多少目录
  
5. 使用`cp`命令：
  - (9) 将用户主文件夹下的`.bashrc`复制到`/usr`目录，并命名为`bashrc1`
  - (10) 在`/tmp`下创建目录`test`，然后复制该目录的内容到`/usr`
  
6. 使用`mv`命令：
  - (11) 将上例中的文件`bashrc1`移动到`/usr/test`目录
  - (12) 将上例中的`test`目录重命名为`test2`
  
7. 使用`rm`命令：
  - (13) 删除复制的`bashrc1`文件
  - (14) 删除`test2`目录
  
8. 使用`cat`命令：
  - (15) 查看用户主文件夹下的`.bashrc`文件内容
  
9. 使用`tac`命令：
  - (16) 反向查看用户主文件夹下的`.bashrc`文件内容

10. 使用`more`命令:

- (17) 逐页查看用户主文件夹下的`.bashrc`文件内容
- (18) 查看用户主文件夹下的`.bashrc`文件内容的前 20 行
- (19) 查看主文件夹下.bashrc 文件内容, 后面 50 行不显示, 只显示前面几行

11. 使用`tail`命令:

- (20) 查看用户主文件夹下的`.bashrc`文件内容的最后 20 行
- (21) 查看用户主文件夹下的`.bashrc`文件内容, 只列出 50 行之后的数据

12. 使用`touch`命令:

- (22) 在`/tmp`下创建一个名为`hello`的空文件, 并查看其时间
- (23) 修改`hello`文件, 将其日期调整为 5 天前

13. 使用`chown`命令:

- (24) 将`hello`文件的所有者更改为`root`帐号, 并查看其属性

14. 使用`find`命令:

- (25) 查找用户主文件夹下文件名为`.bashrc`的文件

15. 使用`tar`命令:

- (26) 在根目录`/`下创建文件夹`test`, 然后打包成`test.tar.gz`
- (27) 解压缩到`/tmp`目录

16. 使用`grep`命令:

- (28) 从`.bashrc`文件中查找字符串`examples`

17. 配置 Java 环境变量:

- (29) 在`.bashrc`中设置 Java 环境变量
- (30) 查看`JAVA\_HOME`变量的值

(二) 熟悉常用的 Hadoop 操作

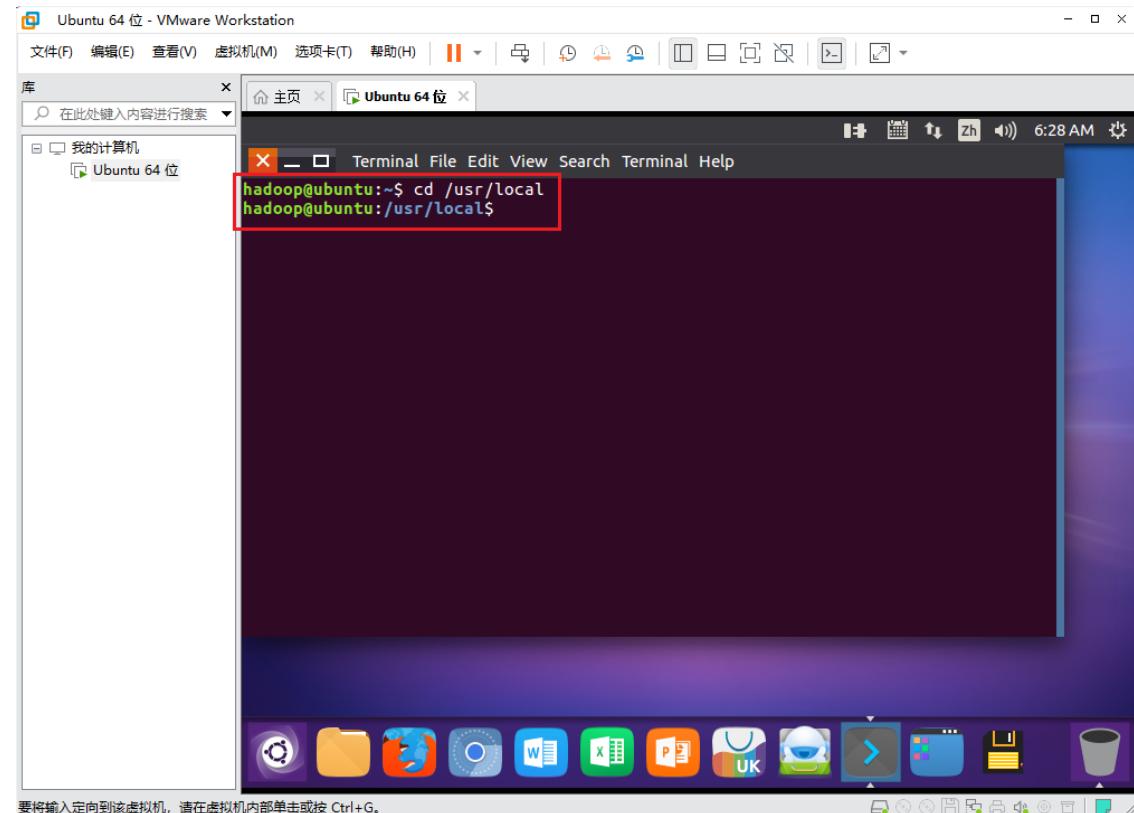
- (31)启动 Hadoop, 并在 Hadoop 分布式文件系统 (HDFS) 中创建用户目录 (假设 Hadoop 已经安装在`/usr/local/hadoop`目录下)。
  - (32)在该用户目录下创建一个名为`text`的文件夹, 并查看 HDFS 中的文件列表。
  - (33)将用户主文件夹下的`.bashrc`文件上传到 HDFS 的`test`文件夹, 并查看`test`文件夹中的内容。
  - (34)将 HDFS 中的`test`文件夹拷贝到本机。

**三、 开发环境:**

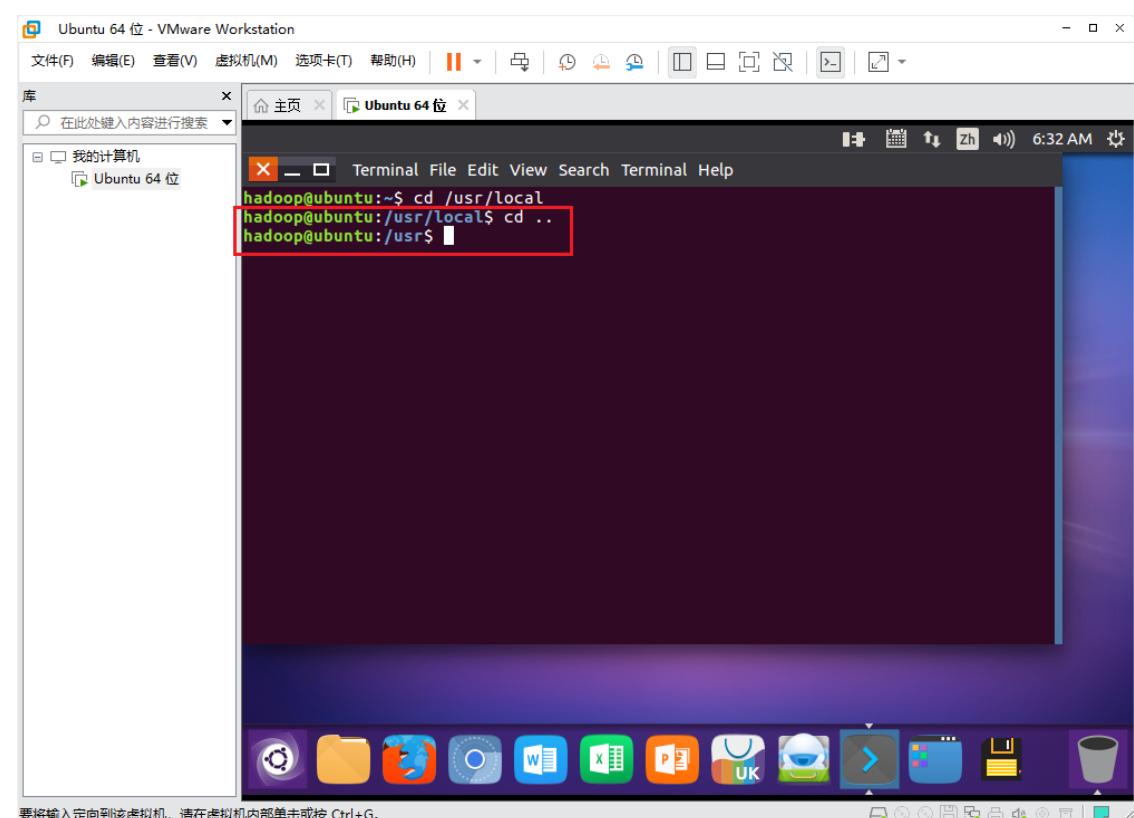
- Lenovo Legion R9000P2021H
- VMWare-workstation-full-17.0.0
- Ubuntu Kylin 16.04
- Eclipse-4.7.0-linux.gtk.x86\_64

#### 四、 实验内容：

(1) 切换到目录`/usr/local`

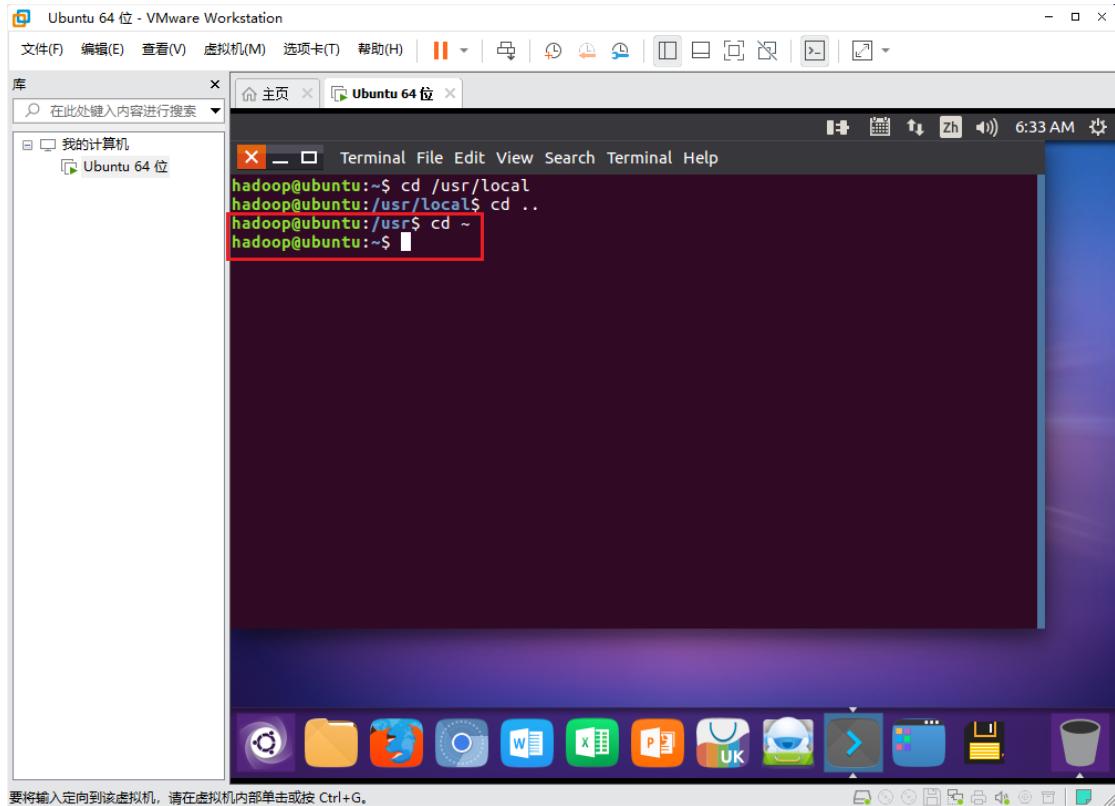


(2) 返回上一级目录

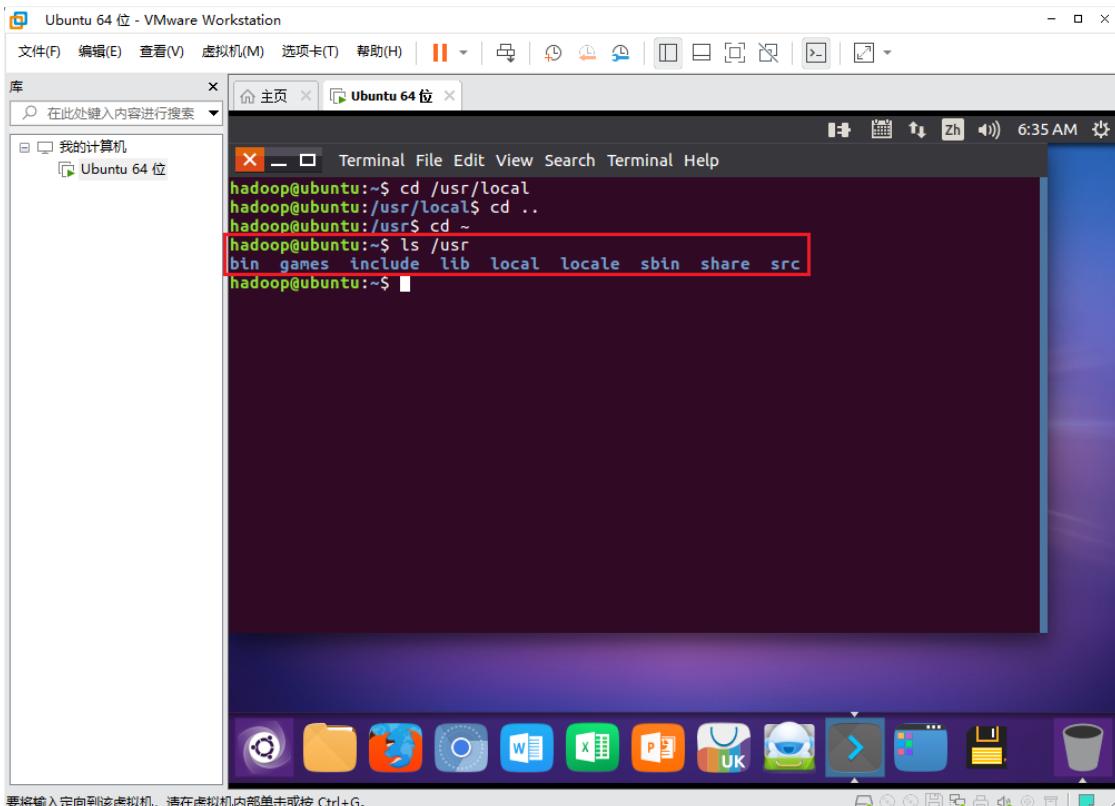


## 实验一

### (3) 返回用户的主文件夹

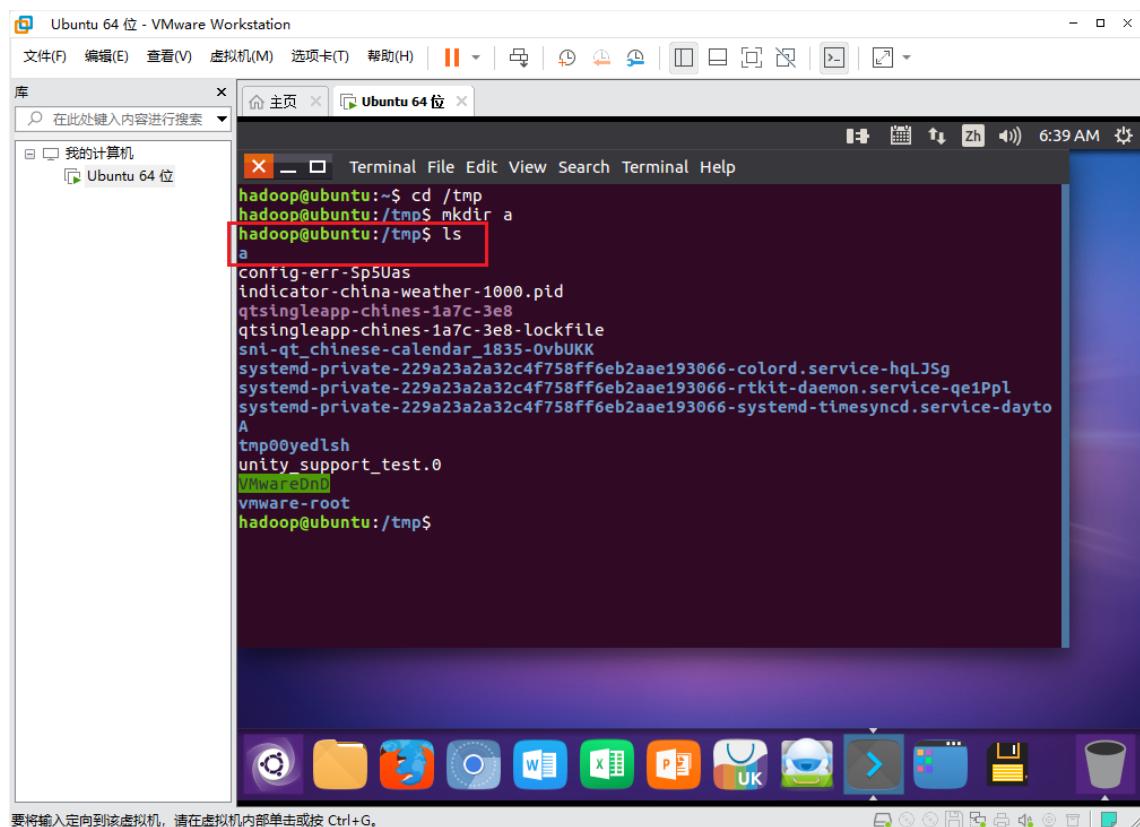
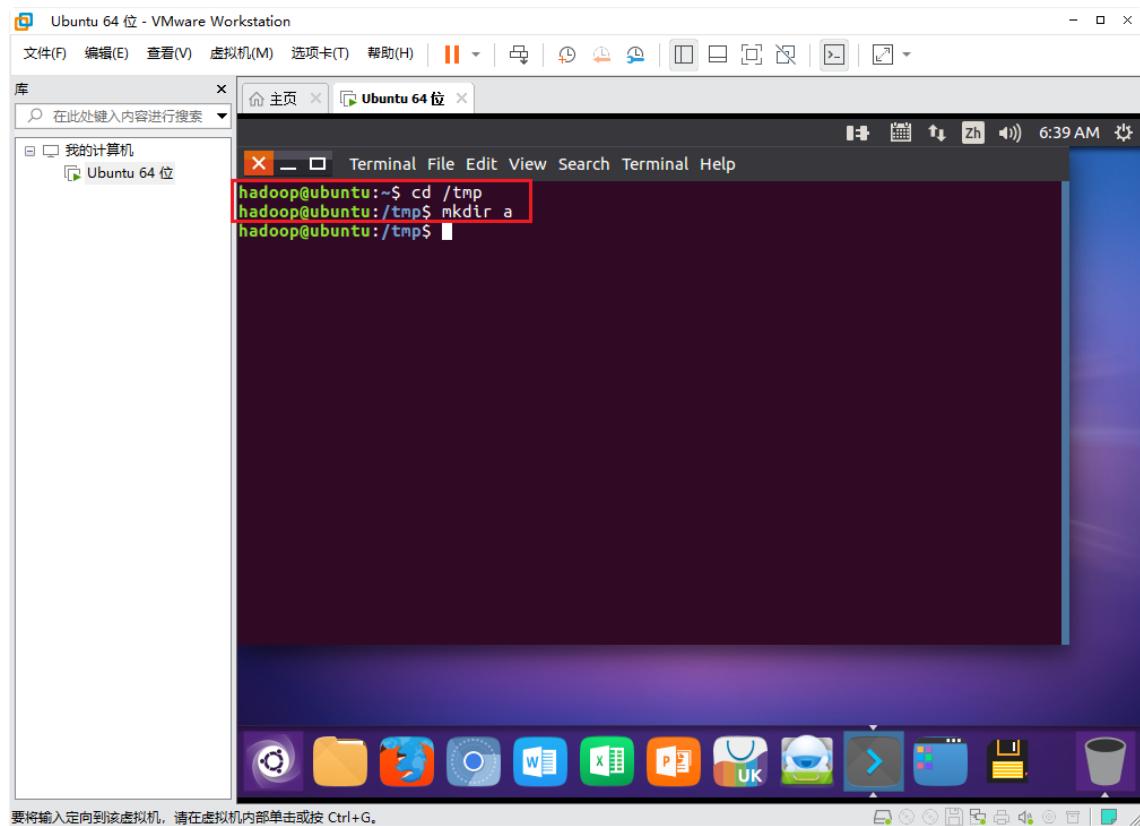


### (4) 查看目录'/usr'下的所有文件



## 实验一

(5) 进入`/tmp`目录，创建一个名为`a`的目录，并查看存在多少目录



## 实验一

(6) 创建目录'a1/a2/a3/a4'

Ubuntu 64 位 - VMware Workstation

文件(F) 编辑(E) 查看(V) 虚拟机(M) 选项卡(T) 帮助(H) | || ▾ ▾ ▾ ▾ ▾ ▾ ▾ ▾ ▾ ▾ ▾ ▾ ▾

库 主页 × Ubuntu 64 位 ×

在此处键入内容进行搜索 ▾

我的计算机 Ubuntu 64 位

Terminal File Edit View Search Terminal Help

```
hadoop@ubuntu:~$ cd /tmp
hadoop@ubuntu:/tmp$ mkdir a
hadoop@ubuntu:/tmp$ ls
a
config-err-Sp5Uas
indicator-china-weather-1000.pid
qtsingleapp-chines-1a7c-3e8
qtsingleapp-chines-1a7c-3e8-lockfile
sni-qt_chinese-calendar_1835-0vBUKK
systemd-private-229a23a2a32c4f758ff6eb2aae193066-colord.service-hqLJ5g
systemd-private-229a23a2a32c4f758ff6eb2aae193066-rtkit-daemon.service-qe1Ppl
systemd-private-229a23a2a32c4f758ff6eb2aae193066-timesyncd.service-dayto
A
tmp0yedlsh
unity_support_test.0
VMwareDnD
vmware-root
hadoop@ubuntu:/tmp$ mkdir -p a1/a2/a3/a4
hadoop@ubuntu:/tmp$
```

(7) 删除在上例中创建的目录'a'（位于'/tmp'下）

## 实验一

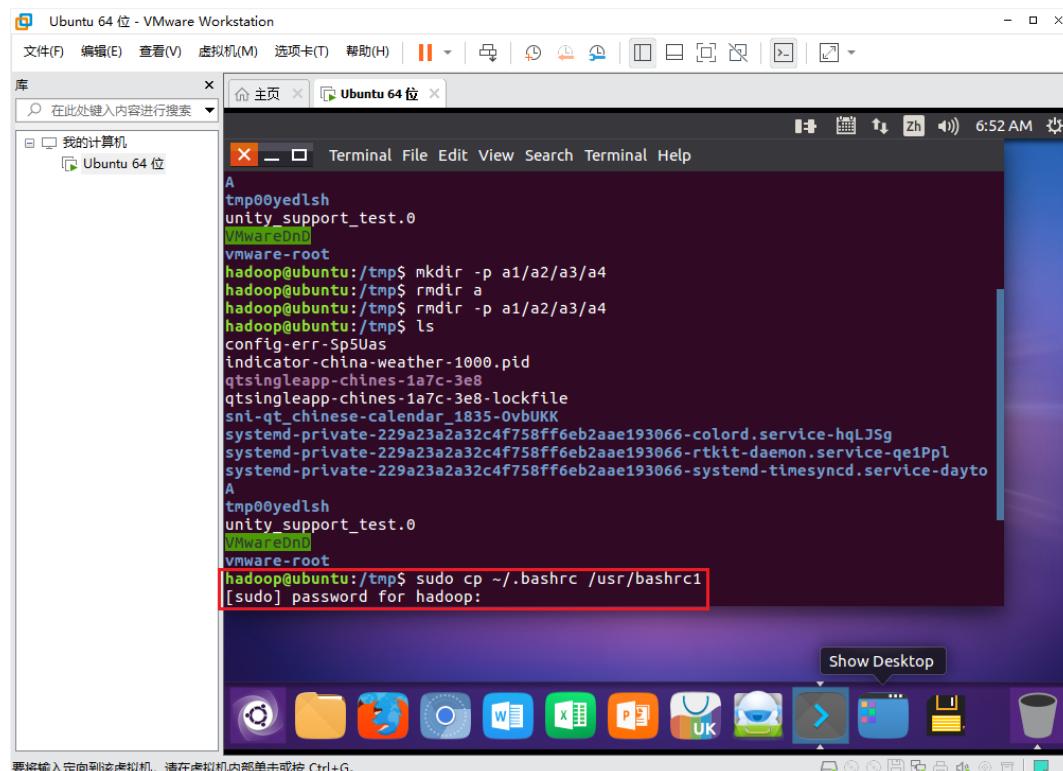
(8) 删除目录'a1/a2/a3/a4'，并查看存在多少目录

The screenshot shows a VMware Workstation interface with an Ubuntu 64-bit virtual machine running. A terminal window is open, displaying the following command history:

```
hadoop@ubuntu:~$ cd /tmp
hadoop@ubuntu:/tmp$ mkdir a
hadoop@ubuntu:/tmp$ ls
a
config-err-Sp5Uas
indicator-china-weather-1000.pid
qtsingleapp-chines-1a7c-3e8
qtsingleapp-chines-1a7c-3e8-lockfile
sni-qt_chinese-calendar_1835-0vbUKK
systemd-private-229a23a2a32c4f758ff6eb2aae193066-colord.service-hqLJSg
systemd-private-229a23a2a32c4f758ff6eb2aae193066-rtkit-daemon.service-qe1Ppl
systemd-private-229a23a2a32c4f758ff6eb2aae193066-timesyncd.service-dayto
A
tmp0yedlsh
unity_support_test.0
VMwareDnD
vmware-root
hadoop@ubuntu:/tmp$ mkdir -p a1/a2/a3/a4
hadoop@ubuntu:/tmp$ rmdir a
hadoop@ubuntu:/tmp$ rmdir -p a1/a2/a3/a4
hadoop@ubuntu:/tmp$
```

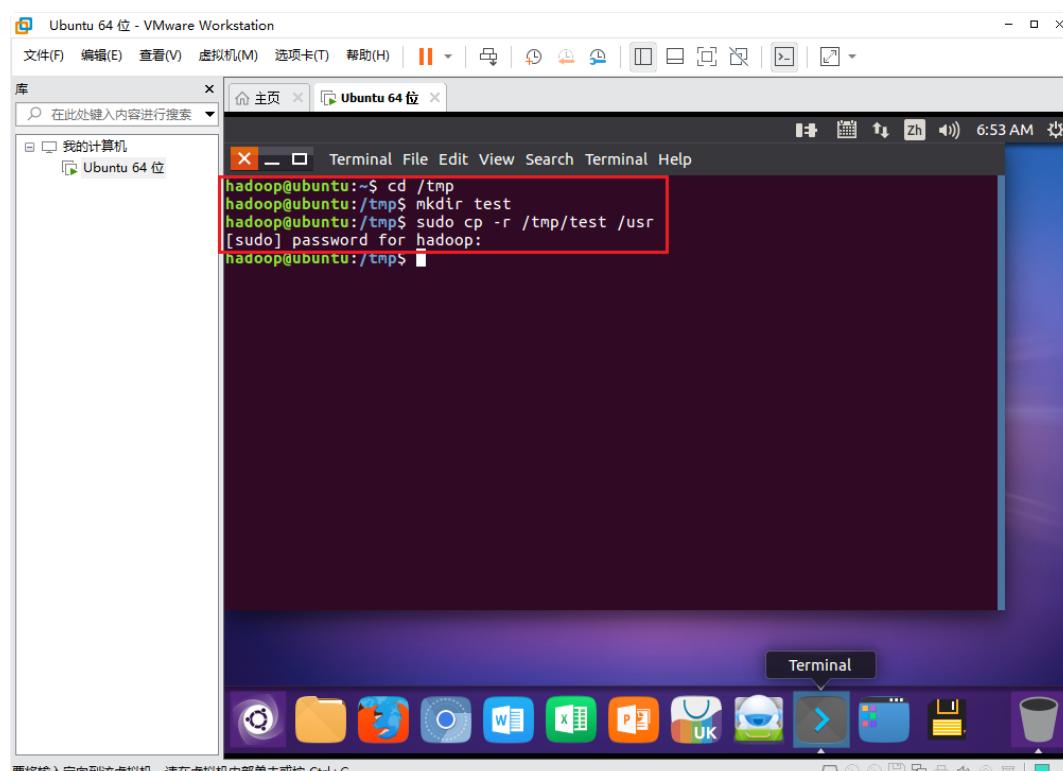
## 实验一

(9) 将用户主文件夹下的`~/.bashrc`复制到`/usr`目录，并命名为`bashrc1`



```
A
tmp0yedlsh
unity_support_test.0
VMware仮想マシン
vmware-root
hadoop@ubuntu:/tmp$ mkdir -p a1/a2/a3/a4
hadoop@ubuntu:/tmp$ rmdir a
hadoop@ubuntu:/tmp$ rmdir -p a1/a2/a3/a4
hadoop@ubuntu:/tmp$ ls
config-err-Sp5Uas
indicator-china-weather-1000.pid
qtsingleapp-chines-1a7c-3e8
qtsingleapp-chines-1a7c-3e8-lockfile
sni-qt_chinese-calendar_1835-0vbUKK
systemd-private-229a23a2a32c4f758ff6eb2aae193066-colord.service-hqLJSg
systemd-private-229a23a2a32c4f758ff6eb2aae193066-rtkit-daemon.service-qe1Ppl
systemd-private-229a23a2a32c4f758ff6eb2aae193066-systemd-timesyncd.service-dayto
A
tmp0yedlsh
unity_support_test.0
VMware仮想マシン
vmware-root
hadoop@ubuntu:/tmp$ sudo cp ~/.bashrc /usr/bashrc1
[sudo] password for hadoop:
```

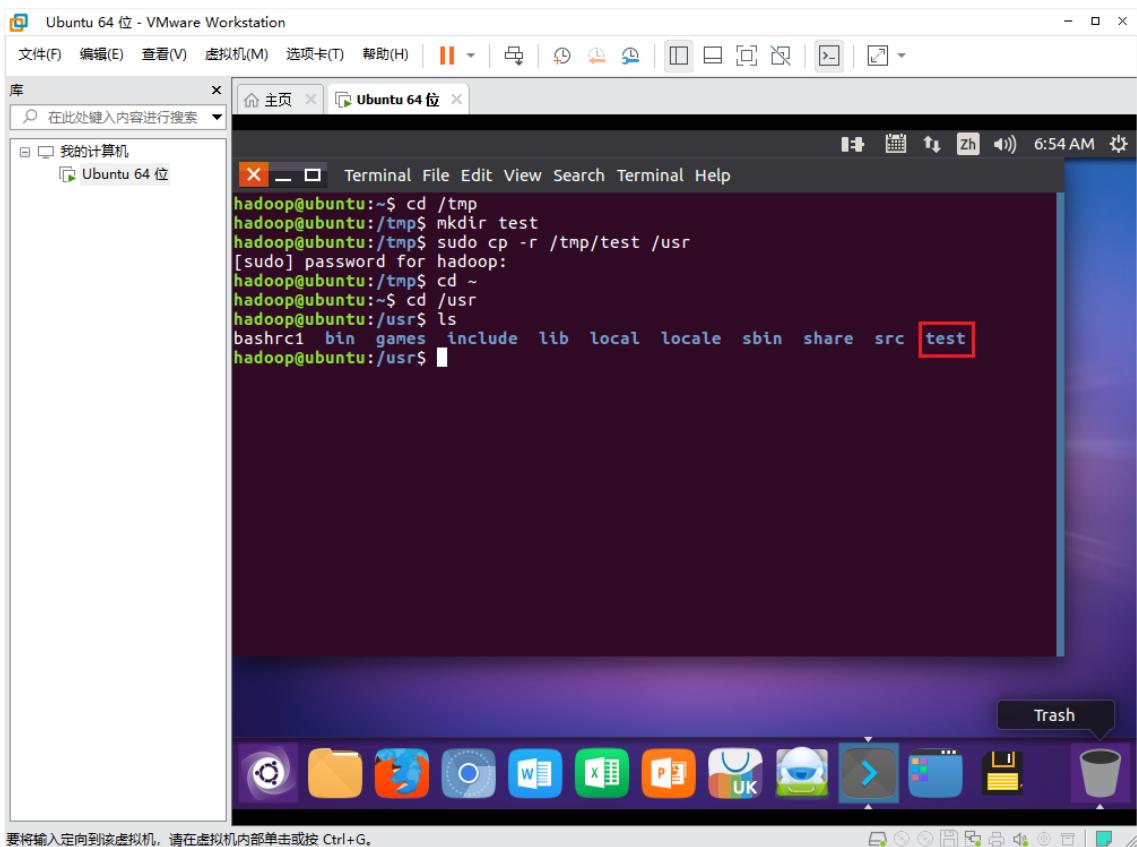
(10) 在`/tmp`下创建目录`test`，然后复制该目录的内容到`/usr`



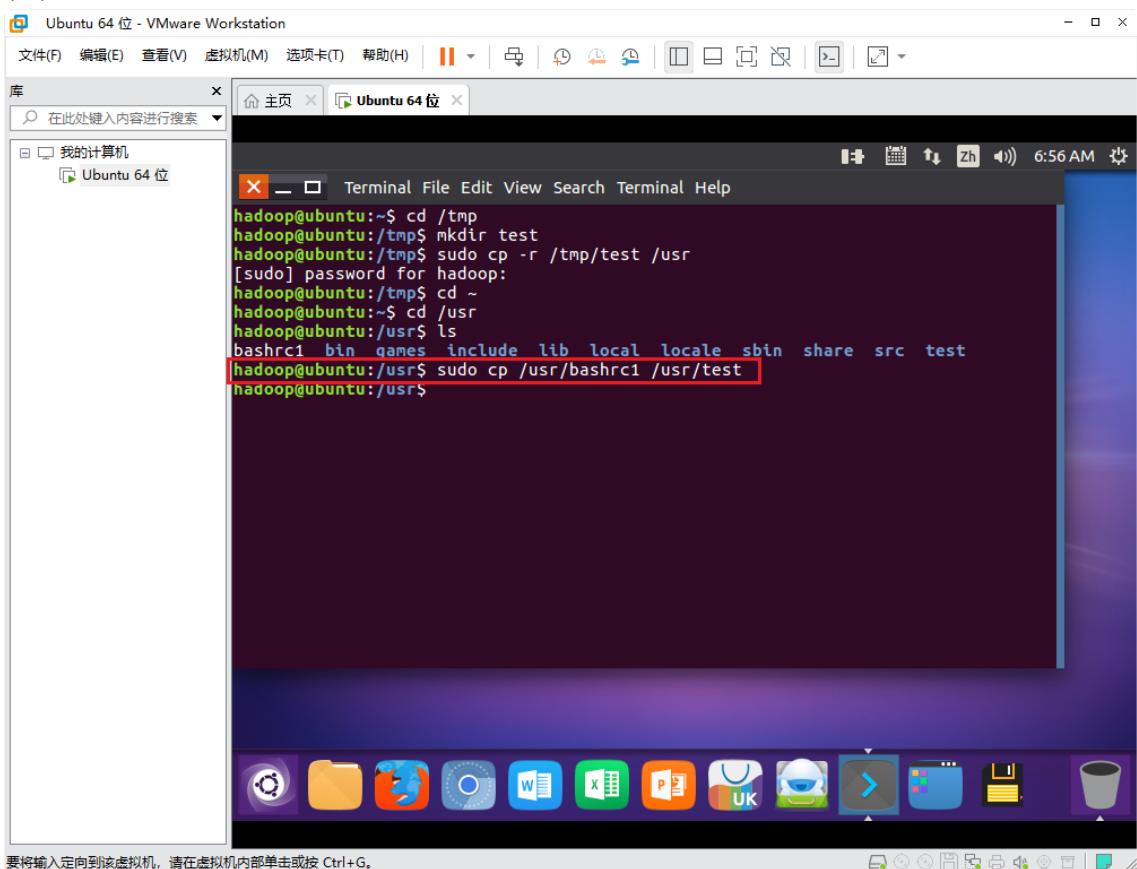
```
hadoop@ubuntu:~/tmp$ cd ~
hadoop@ubuntu:~$ cd /usr
hadoop@ubuntu:/usr$ ls
bashrc1 bin games include lib local locale sbin share src
hadoop@ubuntu:/usr$
```

```
hadoop@ubuntu:~$ cd /tmp
hadoop@ubuntu:/tmp$ mkdir test
hadoop@ubuntu:/tmp$ sudo cp -r /tmp/test /usr
[sudo] password for hadoop:
```

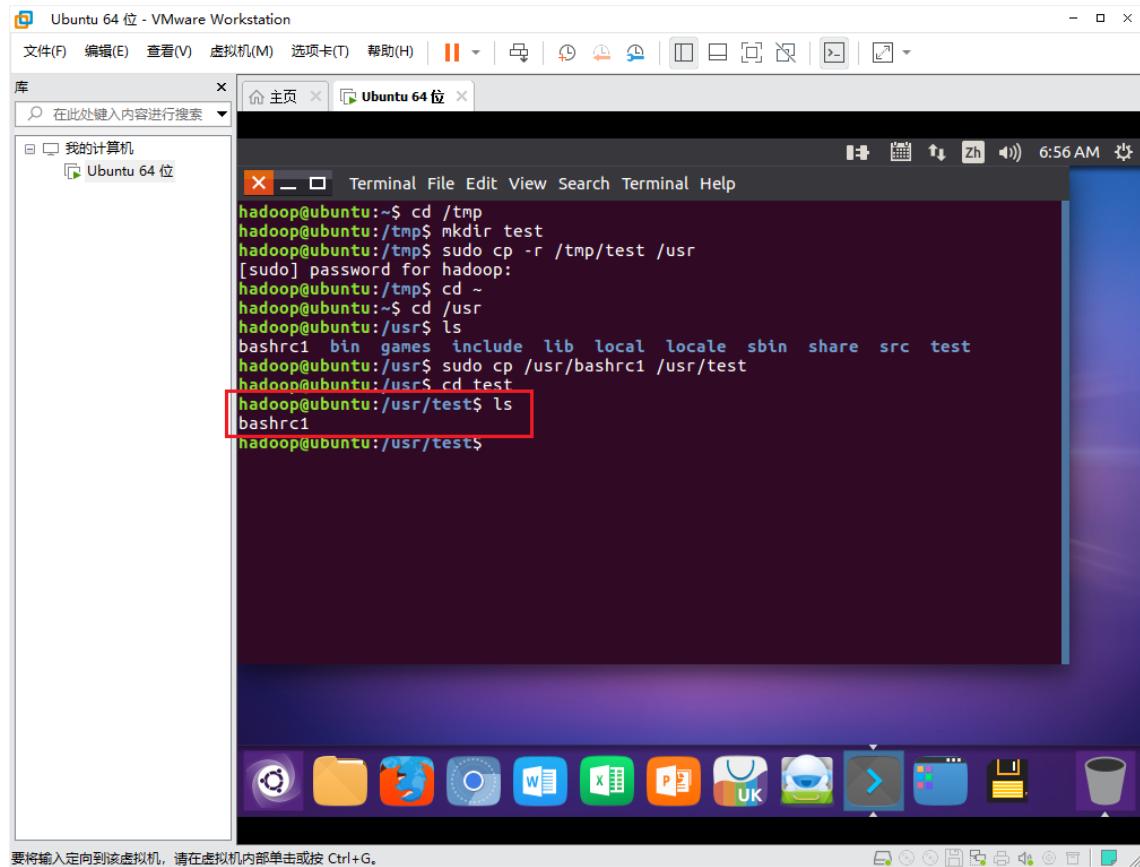
## 实验一



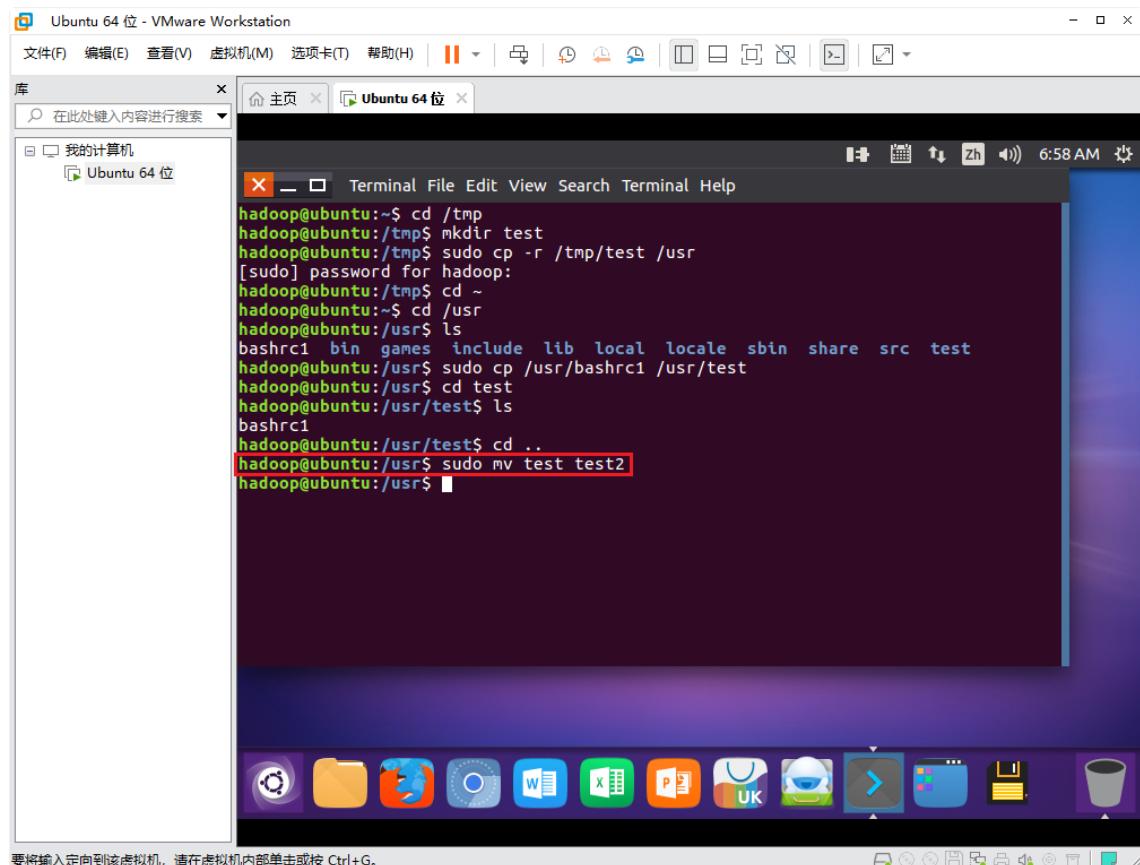
(11) 将上例中的文件`bashrc1`移动到`/usr/test`目录



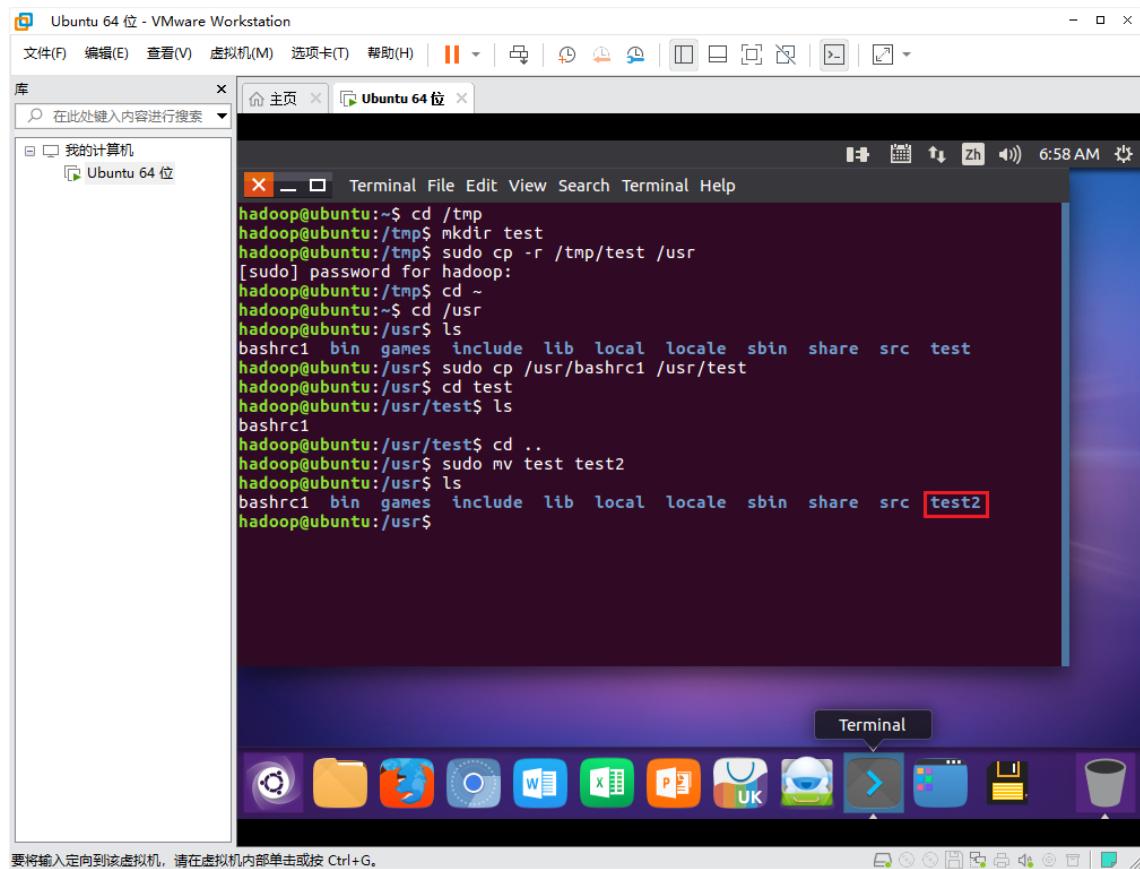
## 实验一



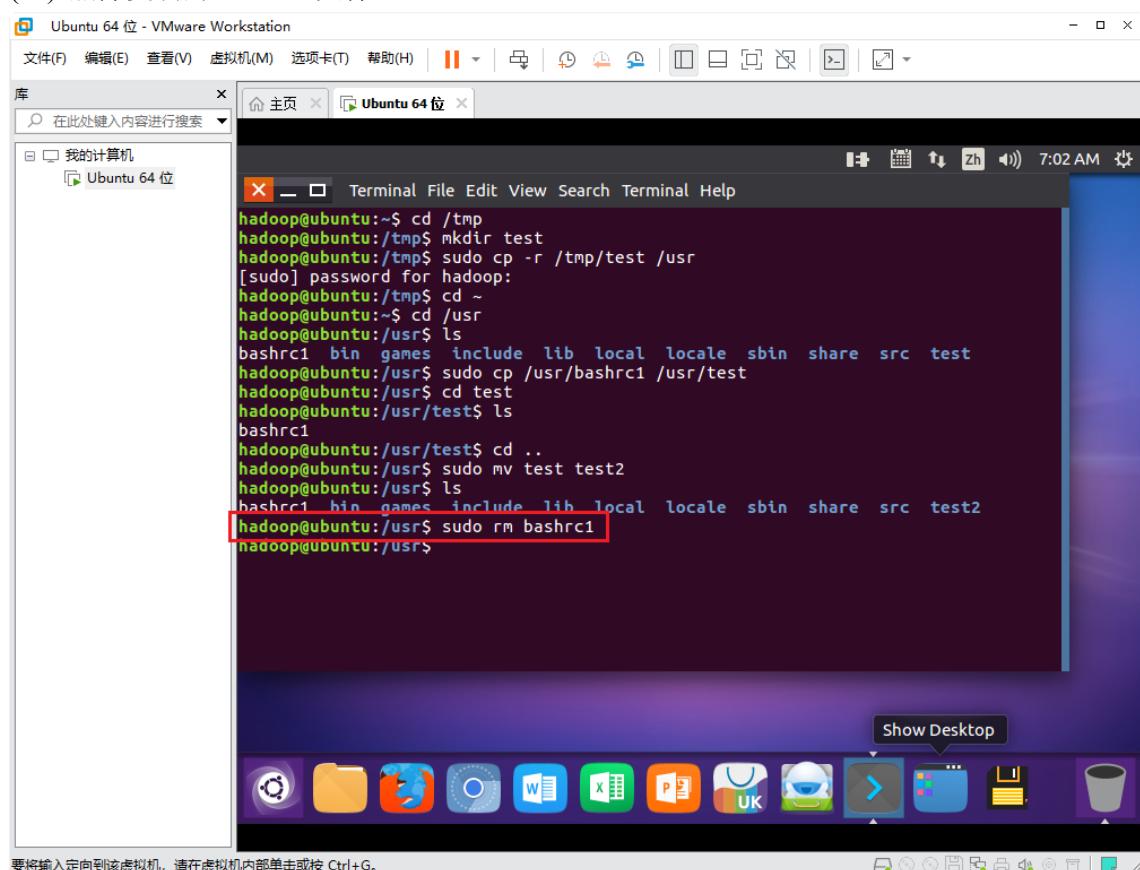
(12) 将上例中的`test`目录重命名为`test2`



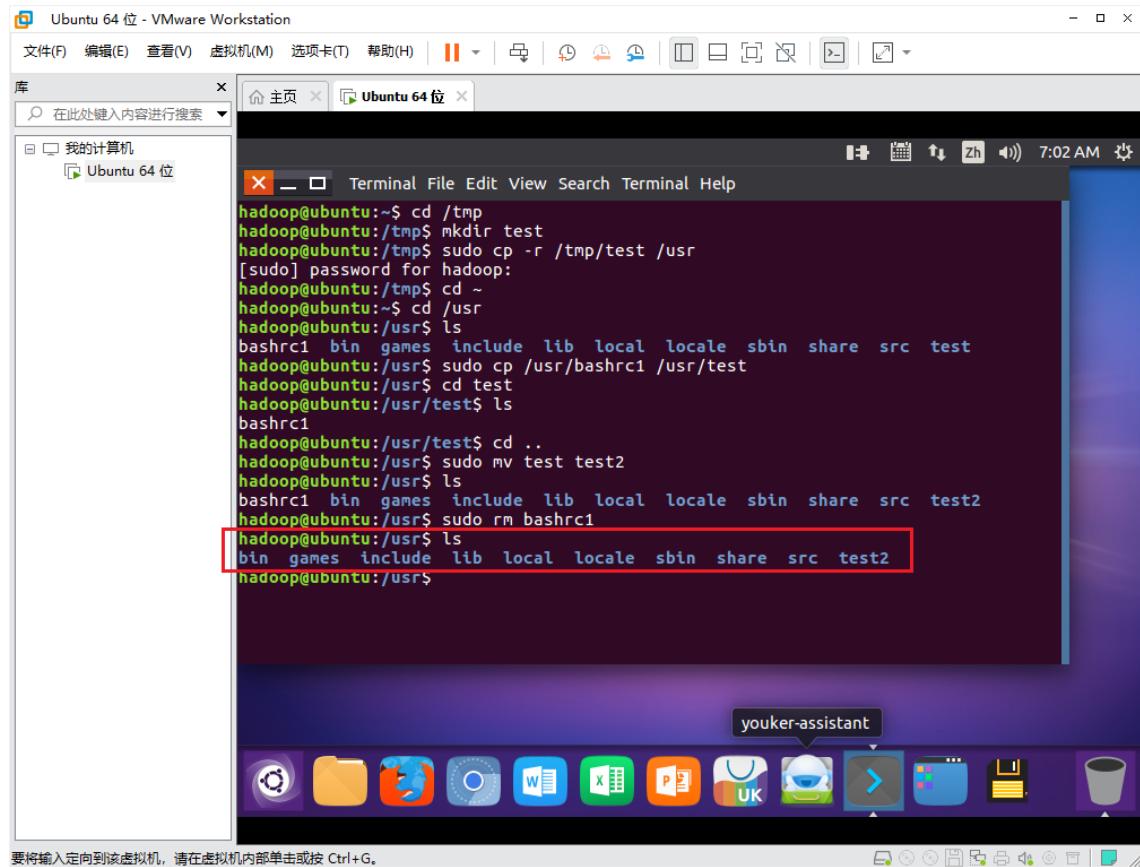
## 实验一



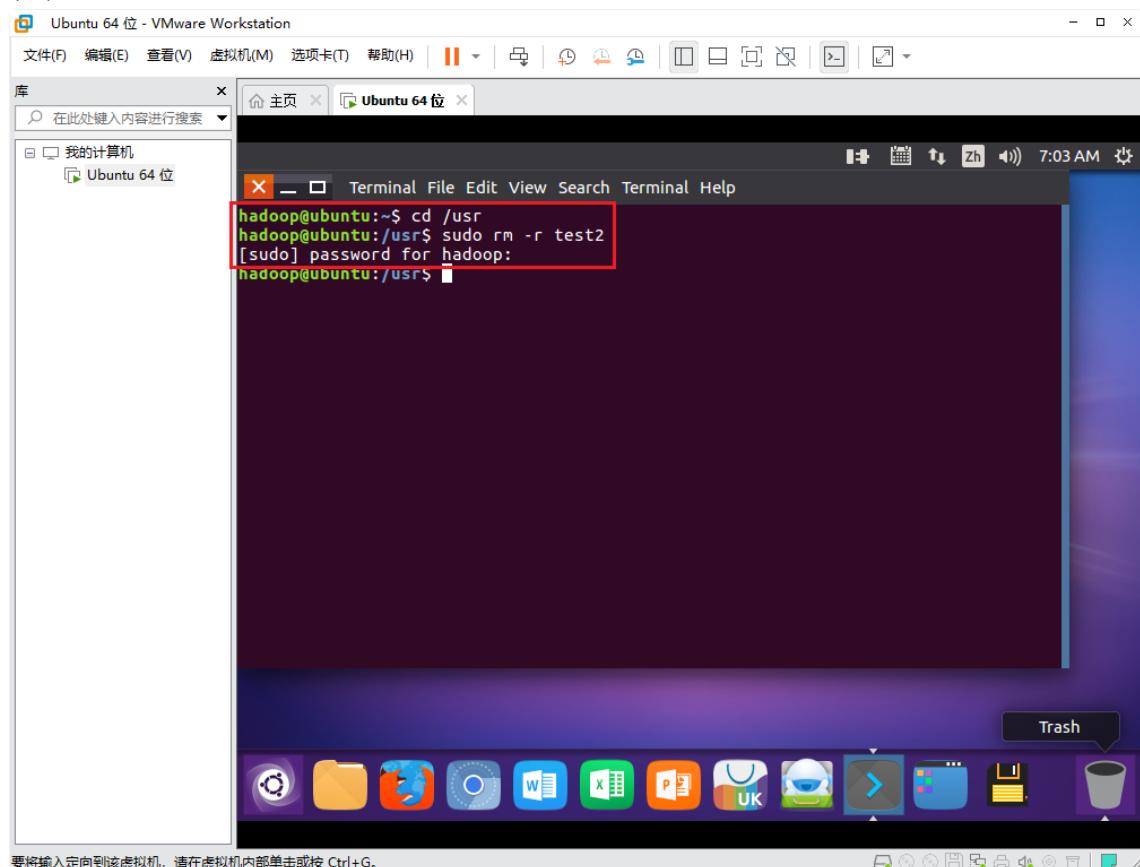
### (13) 删除复制的`bashrc1`文件



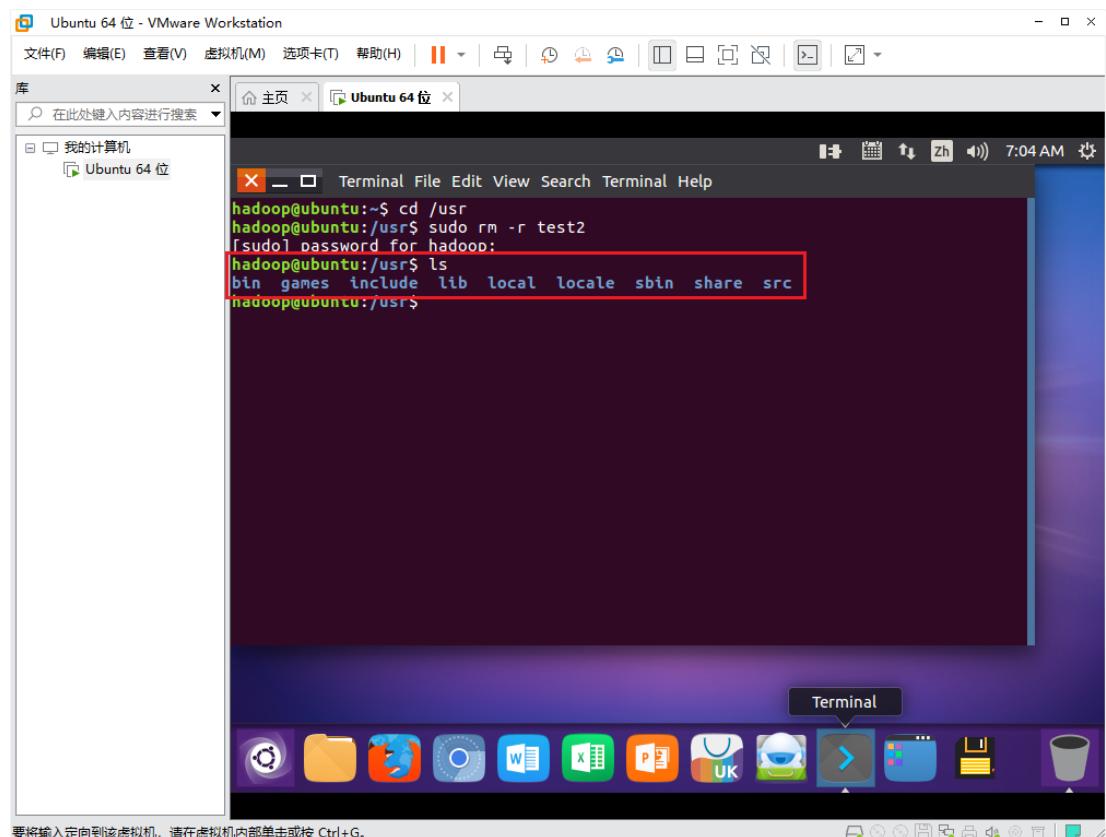
## 实验一



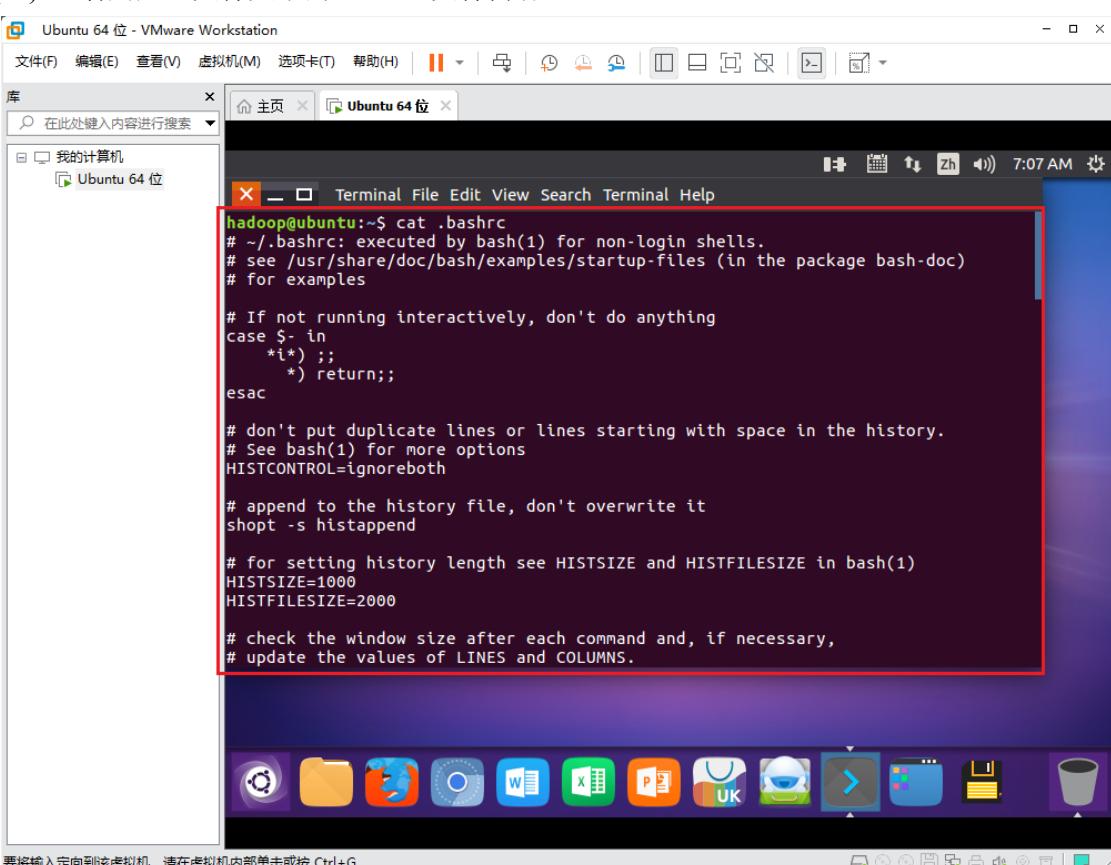
### (14) 删除`test2`目录



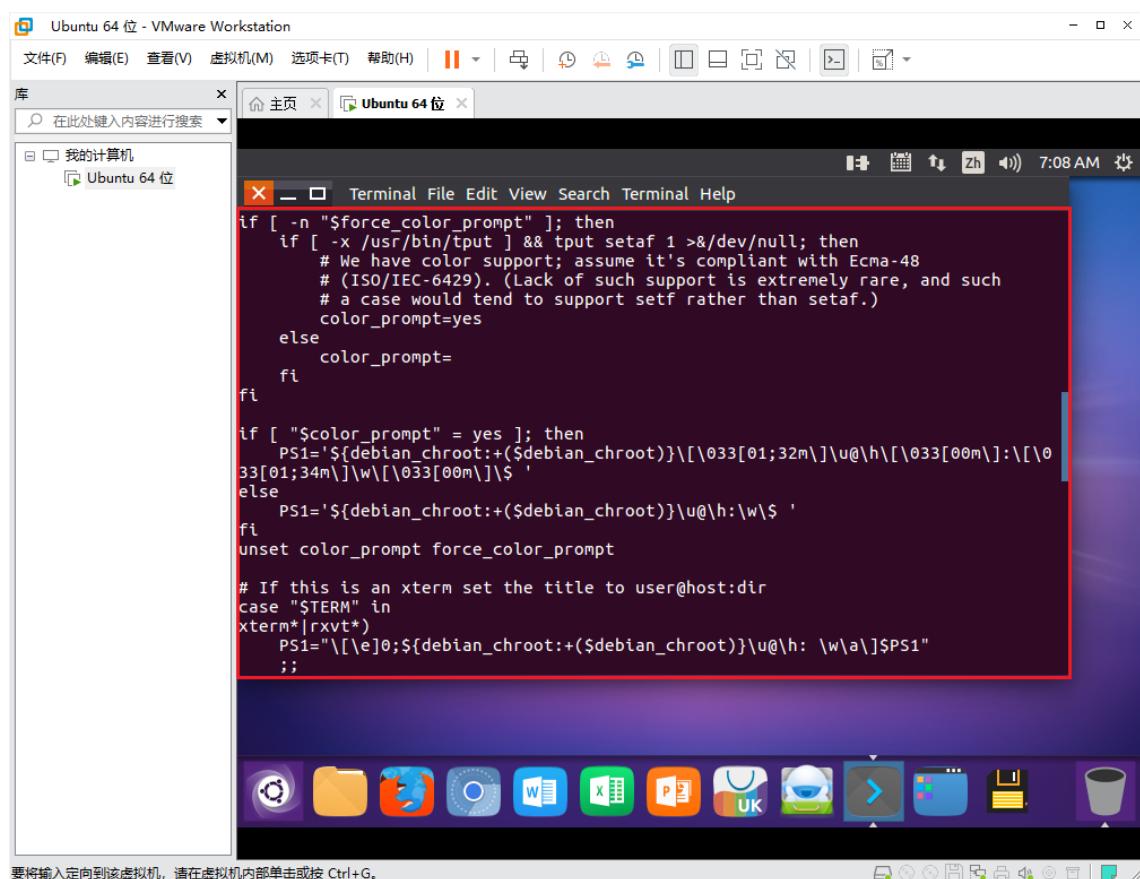
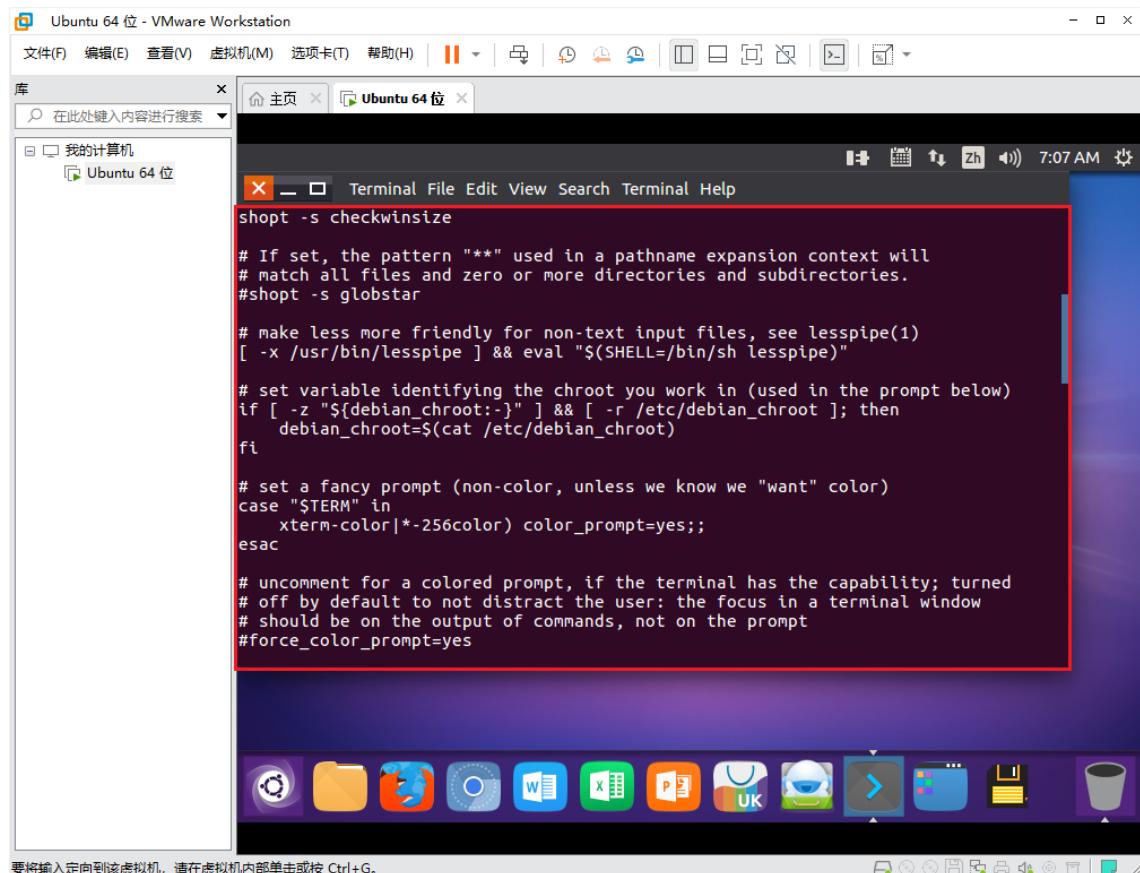
## 实验一



### (15) 查看用户主文件夹下的`.bashrc`文件内容



## 实验一

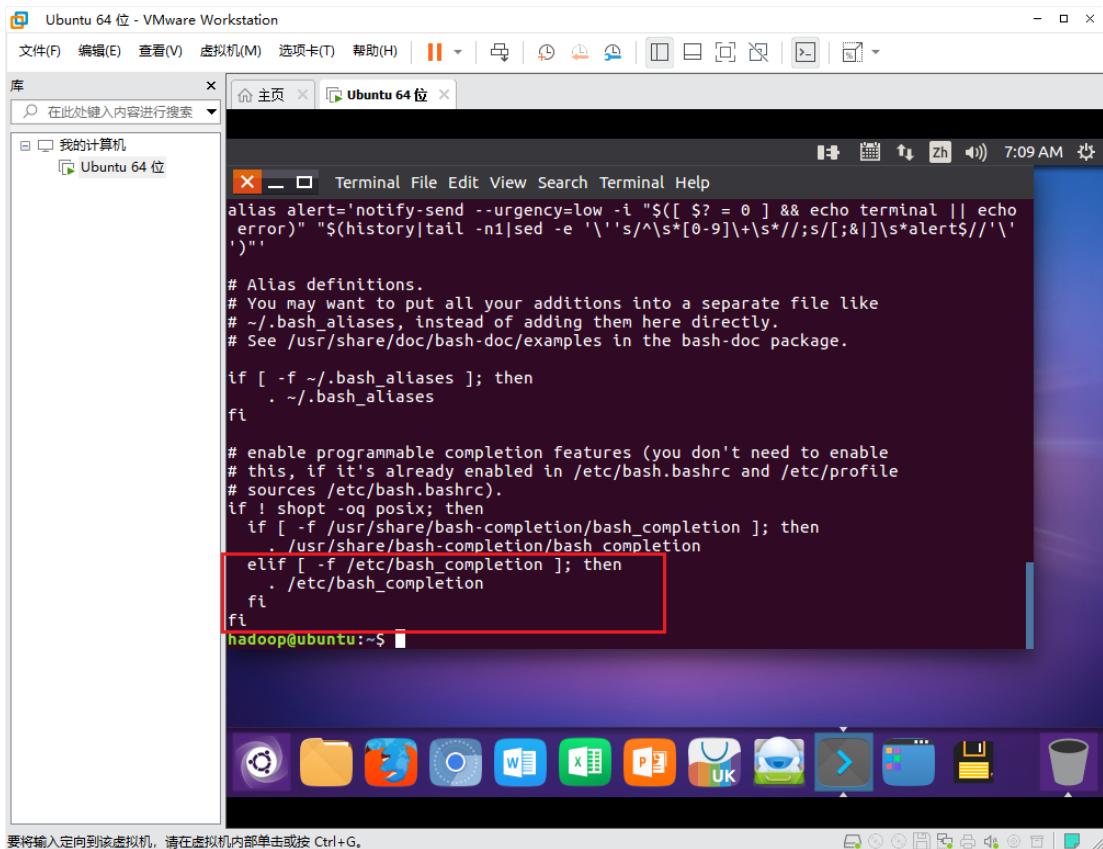


## 实验一

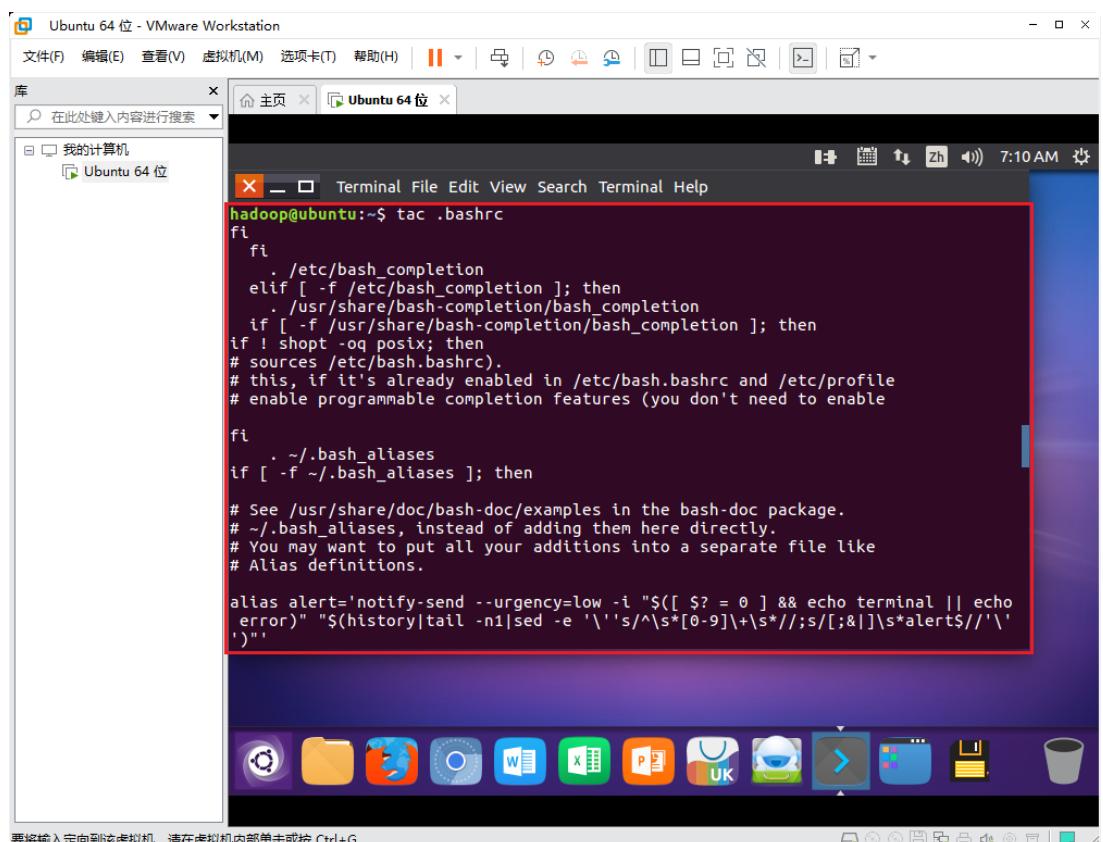
The image displays two screenshots of a VMware Workstation interface, each showing a terminal window on an Ubuntu 64-bit virtual machine. The top screenshot shows the contents of the `/etc/bashrc` file, which includes various alias definitions and color support configurations. The bottom screenshot shows the contents of the `/etc/profile` file, which also contains alias definitions and other system configuration scripts. Both terminals are running on a desktop environment with a purple gradient background and a dock of icons at the bottom.

```
*)  
;;  
esac  
  
# enable color support of ls and also add handy aliases  
if [ -x /usr/bin/dircolors ]; then  
    test -r ~/.dircolors && eval "$(dircolors -b ~/.dircolors)" || eval "$(dircolors -b)"  
    alias ls='ls --color=auto'  
    #alias dir='dir --color=auto'  
    #alias vdir='vdir --color=auto'  
  
    alias grep='grep --color=auto'  
    alias fgrep='fgrep --color=auto'  
    alias egrep='egrep --color=auto'  
fi  
  
# colored GCC warnings and errors  
#export GCC_COLORS='error=01;31:warning=01;35:note=01;36:caret=01;32:locus=01:quote=01'  
  
# some more ls aliases  
alias ll='ls -alF'  
alias la='ls -A'  
  
alias la='ls -A'  
alias l='ls -CF'  
  
# Add an "alert" alias for long running commands. Use like so:  
# sleep 10; alert  
alias alert='notify-send --urgency=low -i "$(($? == 0) && echo terminal || echo error)" "$(history|tail -n1|sed -e '\''$s/^\s*[0-9]+\s*\//;s/[;&]\s*alert$/'\')"'  
  
# Alias definitions.  
# You may want to put all your additions into a separate file like  
# ~/.bash_aliases, instead of adding them here directly.  
# See /usr/share/doc/bash-doc/examples in the bash-doc package.  
  
if [ -f ~/.bash_aliases ]; then  
    . ~/.bash_aliases  
fi  
  
# enable programmable completion features (you don't need to enable  
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile  
# sources /etc/bash.bashrc).  
if ! shopt -oq posix; then  
    if [ -f /usr/share/bash-completion/bash_completion ]; then  
        . /usr/share/bash-completion/bash_completion  
    fi
```

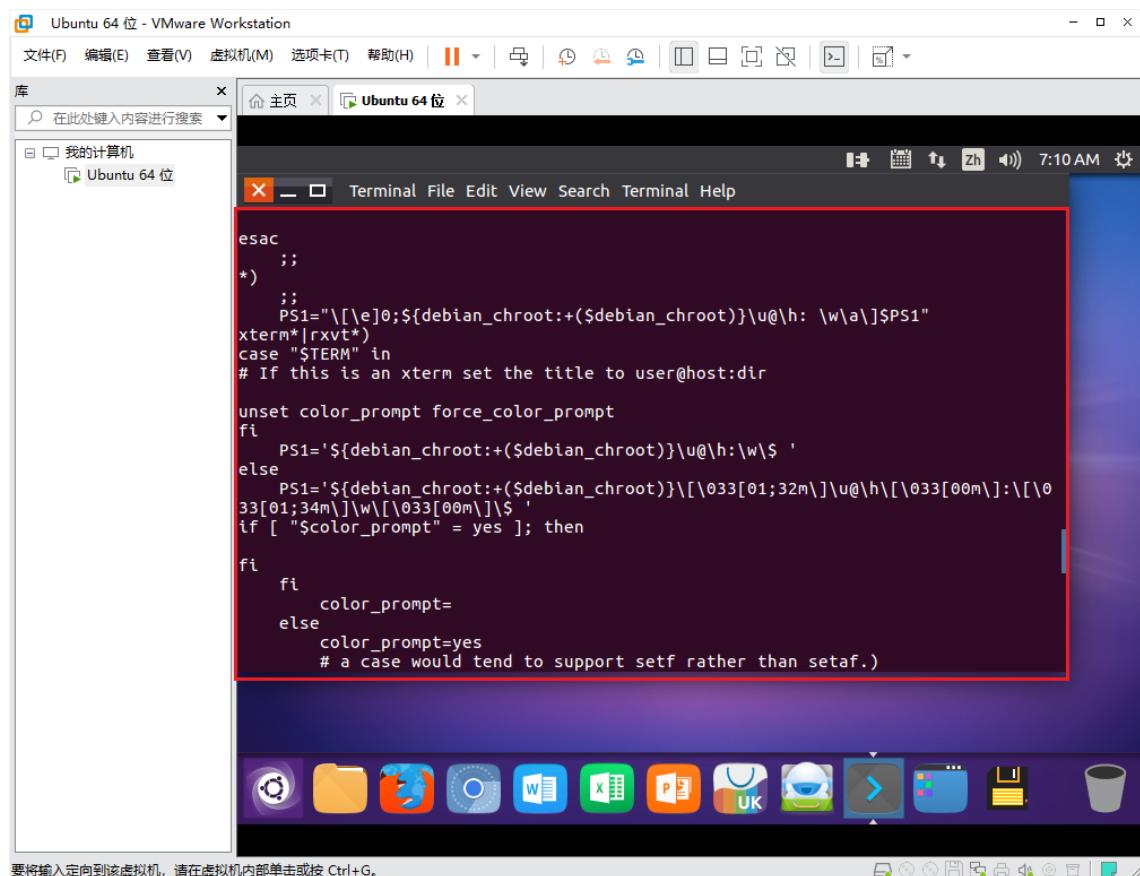
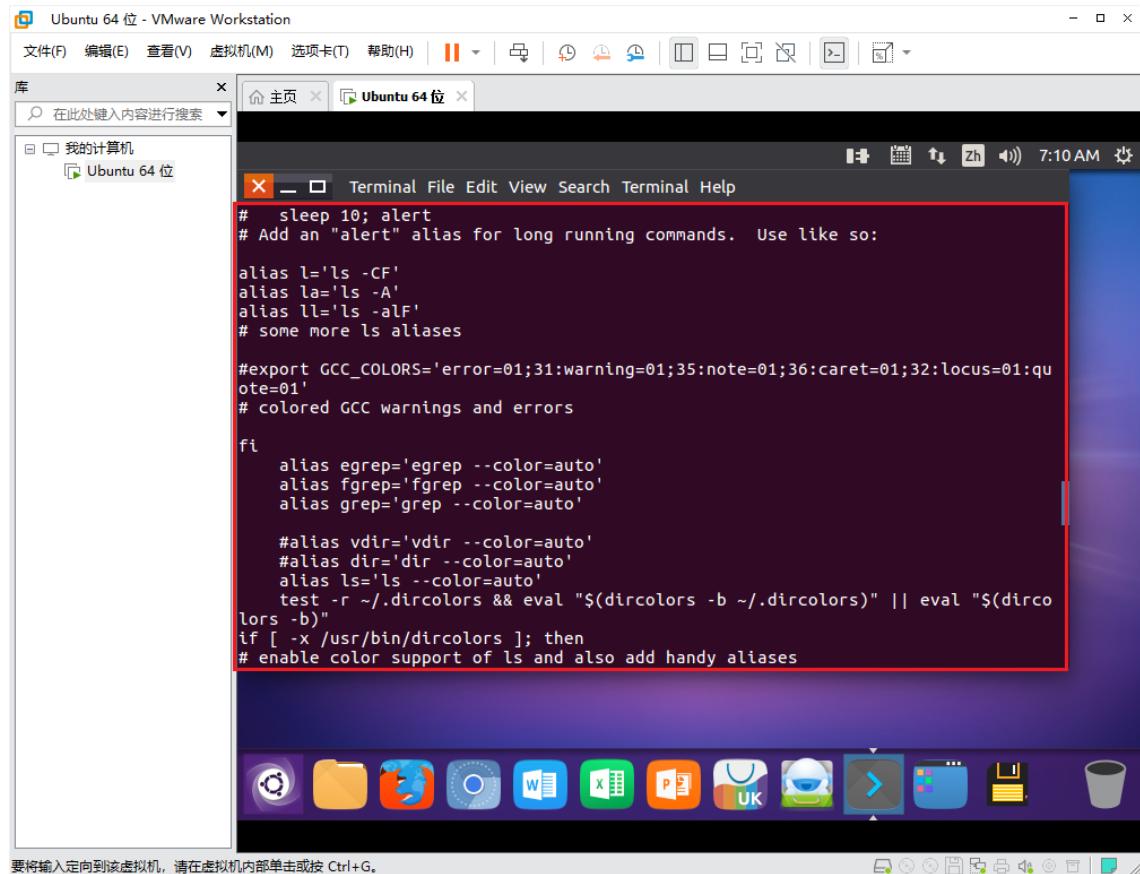
## 实验一



### (16) 反向查看用户主文件夹下的`.bashrc`文件内容



## 实验一



## 实验一

The image displays two screenshots of a VMware Workstation interface, each showing a terminal window on an Ubuntu 64-bit virtual machine. The terminals are running the /etc/bashrc script.

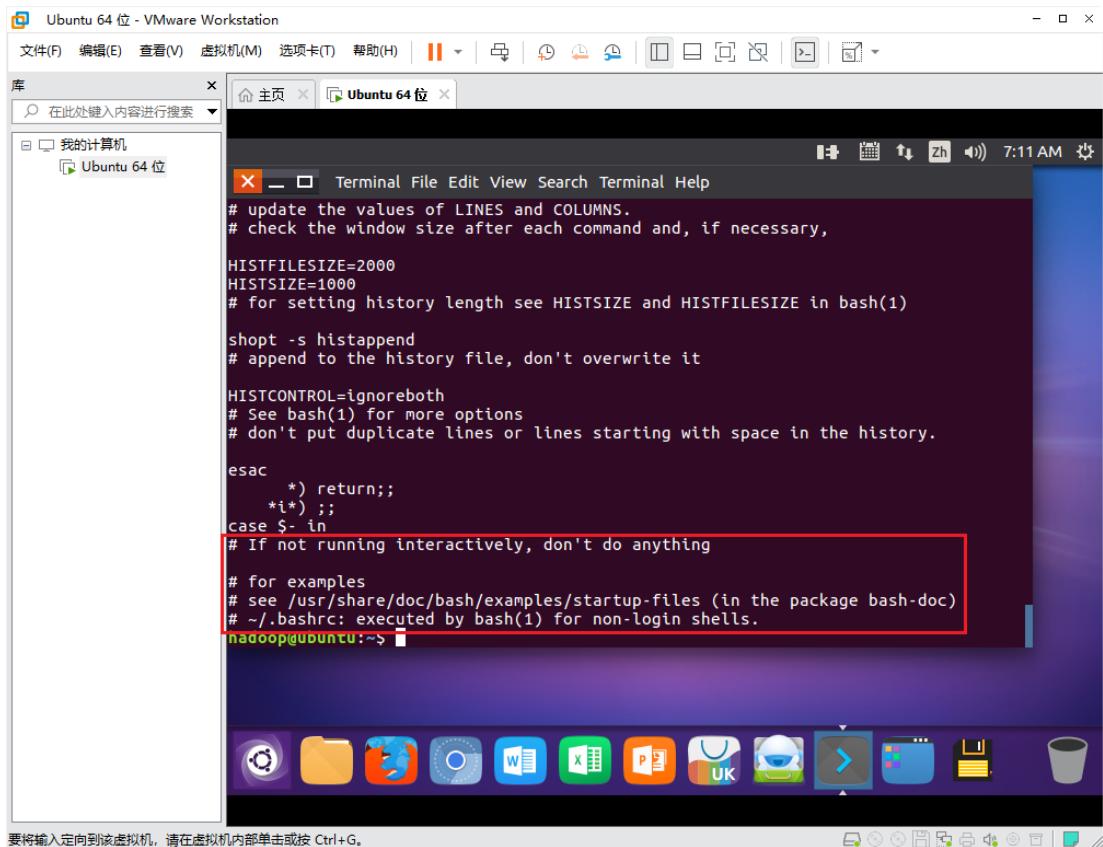
**Top Terminal Session:**

```
# (ISO/IEC-6429). (Lack of such support is extremely rare, and such
# We have color support; assume it's compliant with Ecma-48
if [ -x /usr/bin/tput ] && tput setaf 1 >& /dev/null; then
if [ -n "$force_color_prompt" ]; then
#force_color_prompt=yes
# should be on the output of commands, not on the prompt
# off by default to not distract the user: the focus in a terminal window
# uncomment for a colored prompt, if the terminal has the capability; turned
esac
xterm-color|*-256color) color_prompt=yes;;
case "$TERM" in
# set a fancy prompt (non-color, unless we know we "want" color)
fi
debian_chroot=$(cat /etc/debian_chroot)
if [ -z "$debian_chroot:-:" ] && [ -r /etc/debian_chroot ]; then
# set variable identifying the chroot you work in (used in the prompt below)
[ -x /usr/bin/lesspipe ] && eval "$(SHELL=/bin/sh lesspipe)"
# make less more friendly for non-text input files, see lesspipe(1)
#shopt -s globstar
```

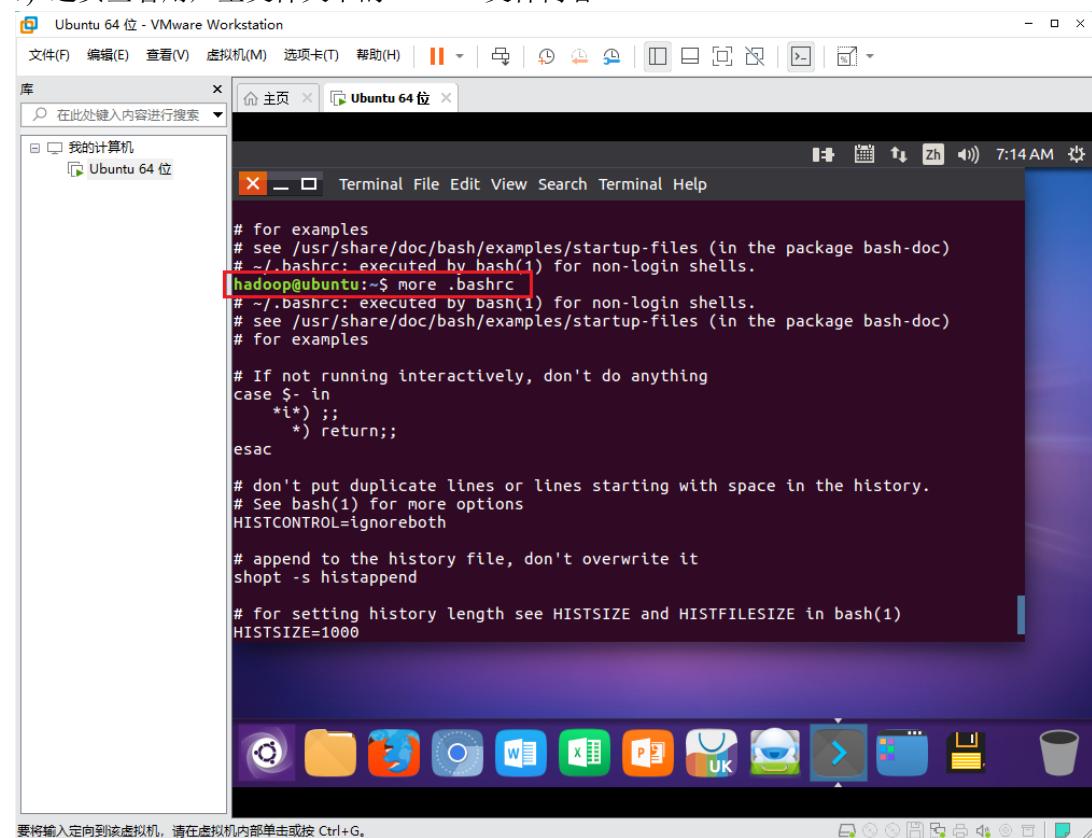
**Bottom Terminal Session:**

```
#shopt -s globstar
# match all files and zero or more directories and subdirectories.
# If set, the pattern "##" used in a pathname expansion context will
shopt -s checkwinsize
# update the values of LINES and COLUMNS.
# check the window size after each command and, if necessary,
HISTFILESIZE=2000
HISTSIZE=1000
# for setting history length see HISTSIZE and HISTFILESIZE in bash(1)
shopt -s histappend
# append to the history file, don't overwrite it
HISTCONTROL=ignoreboth
# See bash(1) for more options
# don't put duplicate lines or lines starting with space in the history.
esac
*) return;;
*i*) ;;
case $- in
```

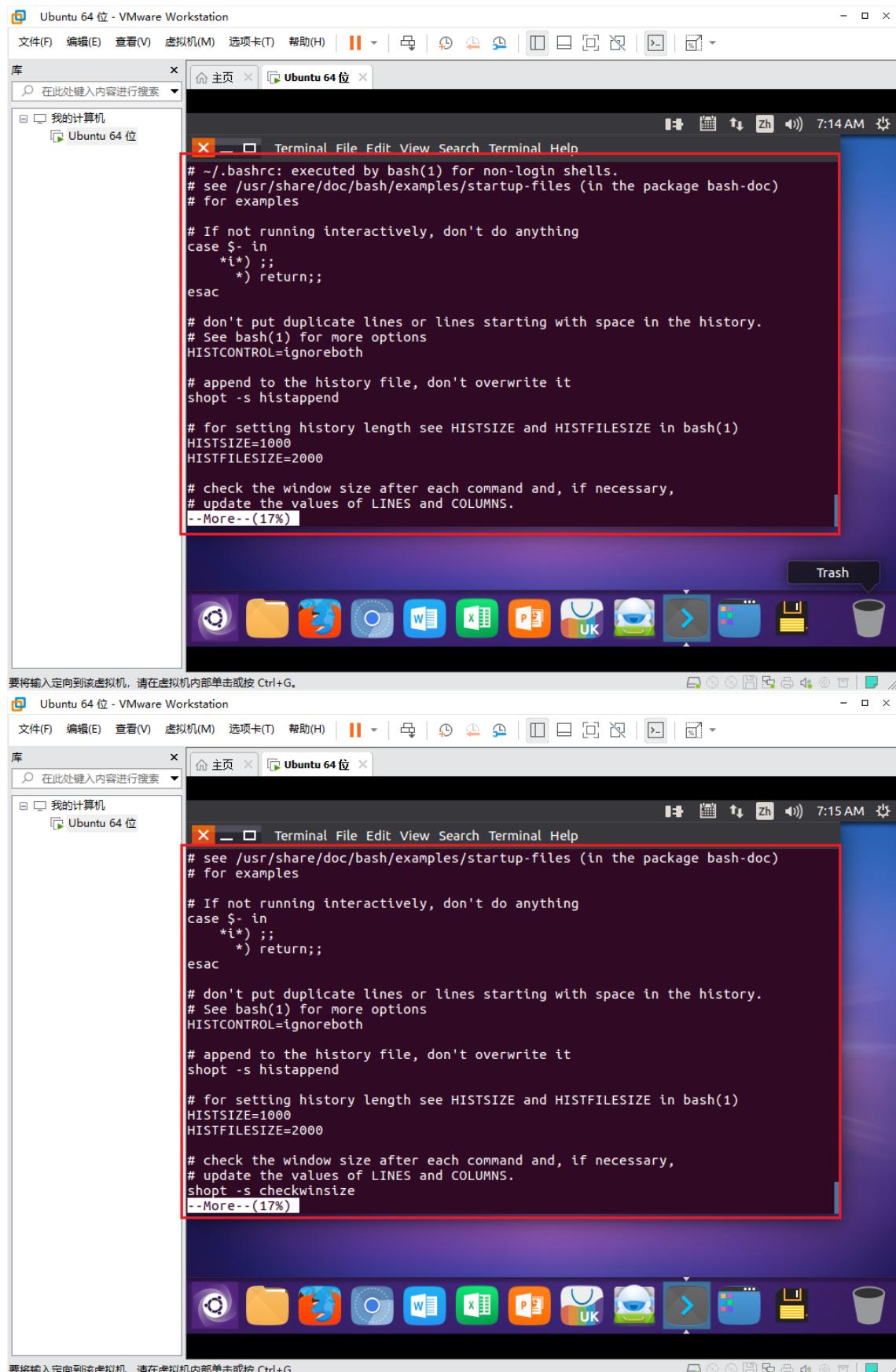
## 实验一



### (17) 逐页查看用户主文件夹下的`.bashrc`文件内容



## 实验一



```
# ~/.bashrc: executed by bash(1) for non-login shells.
# see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
# for examples

# If not running interactively, don't do anything
case $- in
    *i*) ;;
    *) return;;
esac

# don't put duplicate lines or lines starting with space in the history.
# See bash(1) for more options
HISTCONTROL=ignoreboth

# append to the history file, don't overwrite it
shopt -s histappend

# for setting history length see HISTSIZE and HISTFILESIZE in bash(1)
HISTSIZE=1000
HISTFILESIZE=2000

# check the window size after each command and, if necessary,
# update the values of LINES and COLUMNS.
--More--(17%)
```

```
# see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
# for examples

# If not running interactively, don't do anything
case $- in
    *i*) ;;
    *) return;;
esac

# don't put duplicate lines or lines starting with space in the history.
# See bash(1) for more options
HISTCONTROL=ignoreboth

# append to the history file, don't overwrite it
shopt -s histappend

# for setting history length see HISTSIZE and HISTFILESIZE in bash(1)
HISTSIZE=1000
HISTFILESIZE=2000

# check the window size after each command and, if necessary,
# update the values of LINES and COLUMNS.
shopt -s checkwinsize
--More--(17%)
```

## 实验一

The image shows two side-by-side screenshots of a VMware Workstation interface, each displaying a terminal window on an Ubuntu 64-bit virtual machine. The terminals are running the /etc/bash.bashrc script.

**Terminal Content (Left Window):**

```
# for examples
# If not running interactively, don't do anything
case $- in
    *i*) ;;
    *) return;;
esac

# don't put duplicate lines or lines starting with space in the history.
# See bash(1) for more options
HISTCONTROL=ignoreboth

# append to the history file, don't overwrite it
shopt -s histappend

# for setting history length see HISTSIZE and HISTFILESIZE in bash(1)
HISTSIZE=1000
HISTFILESIZE=2000

# check the window size after each command and, if necessary,
# update the values of LINES and COLUMNS.
shopt -s checkwinsize

--More--(17%)
```

**Terminal Content (Right Window):**

```
# for setting history length see HISTSIZE and HISTFILESIZE in bash(1)
HISTSIZE=1000
HISTFILESIZE=2000

# check the window size after each command and, if necessary,
# update the values of LINES and COLUMNS.
shopt -s checkwinsize

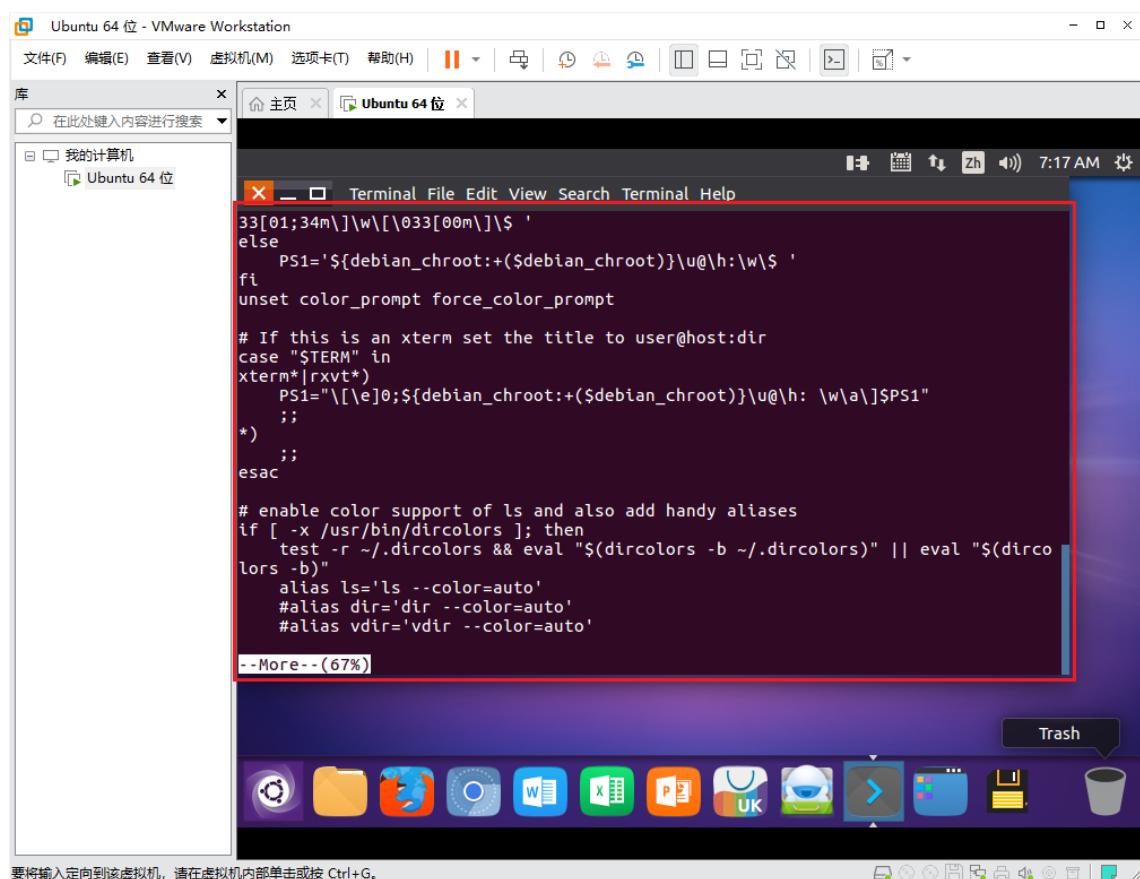
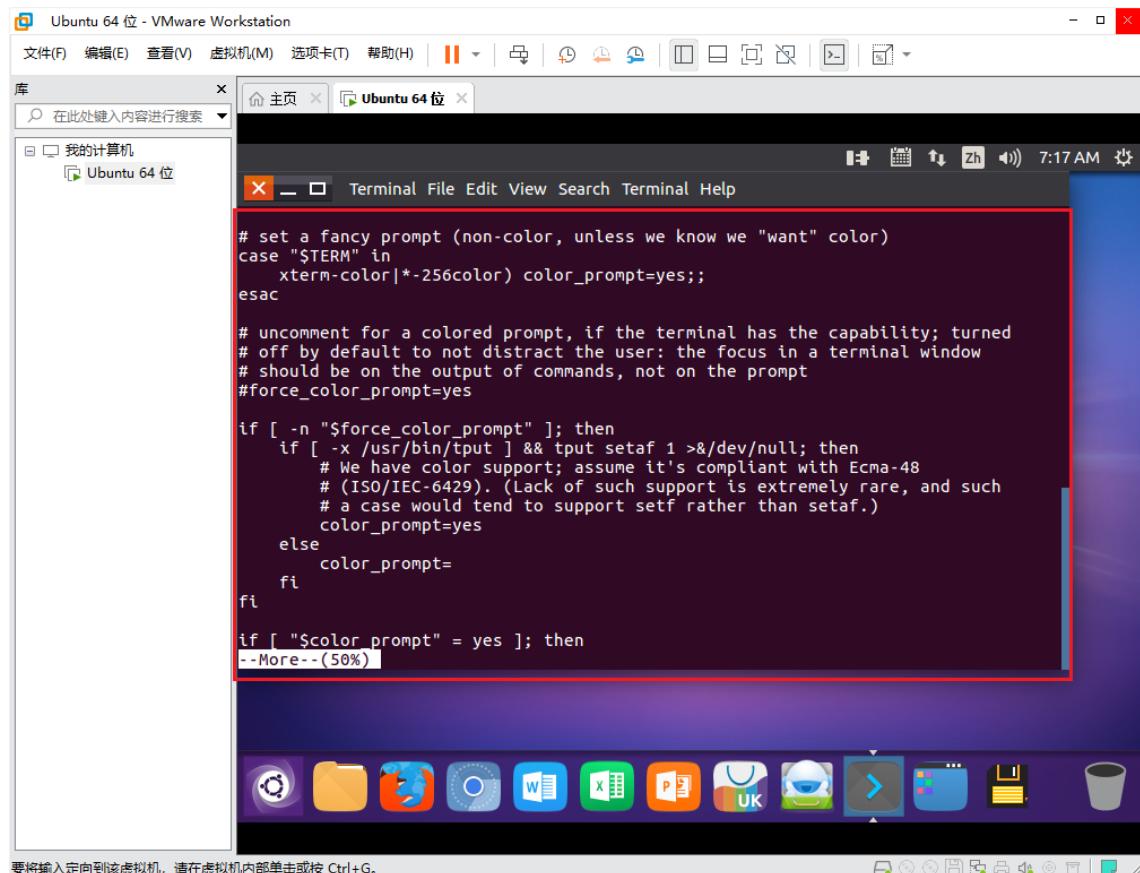
# If set, the pattern "##" used in a pathname expansion context will
# match all files and zero or more directories and subdirectories.
#shopt -s globstar

# make less more friendly for non-text input files, see lesspipe(1)
[ -x /usr/bin/lesspipe ] && eval "$(SHELL=/bin/sh lesspipe)"

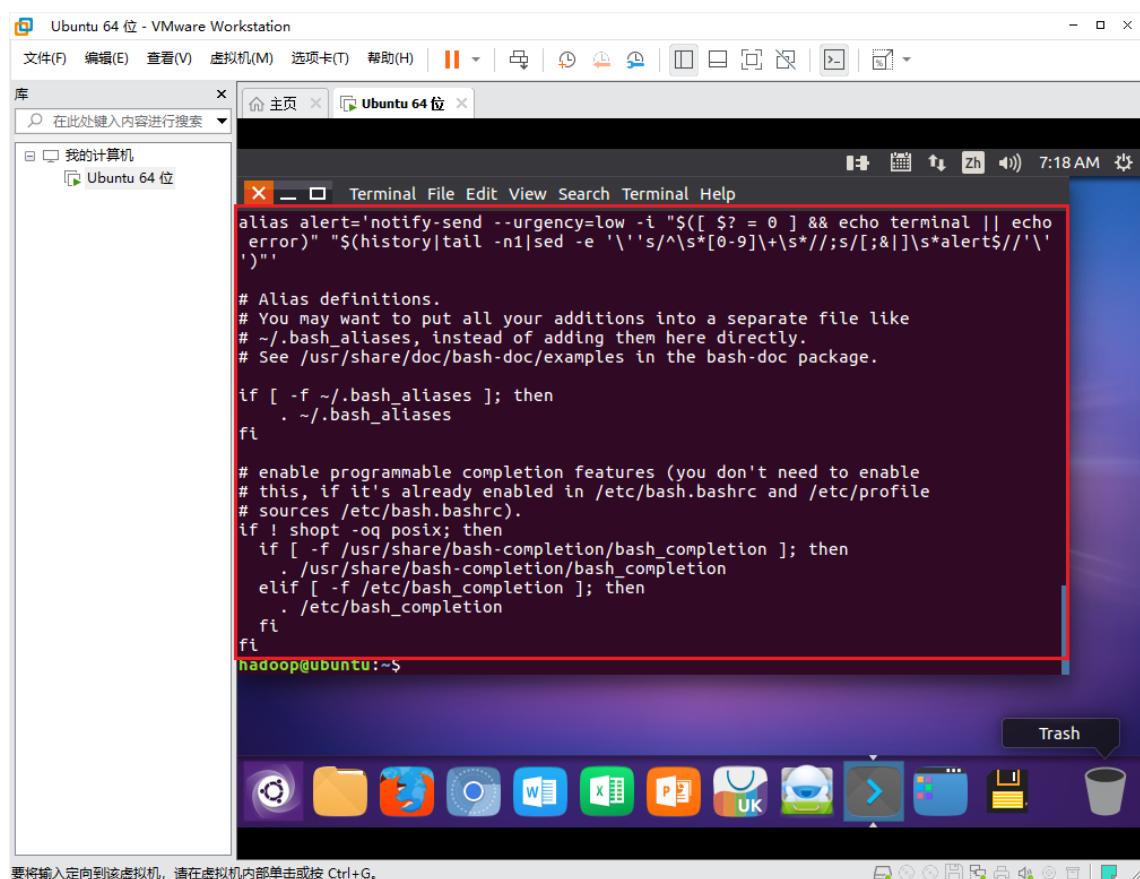
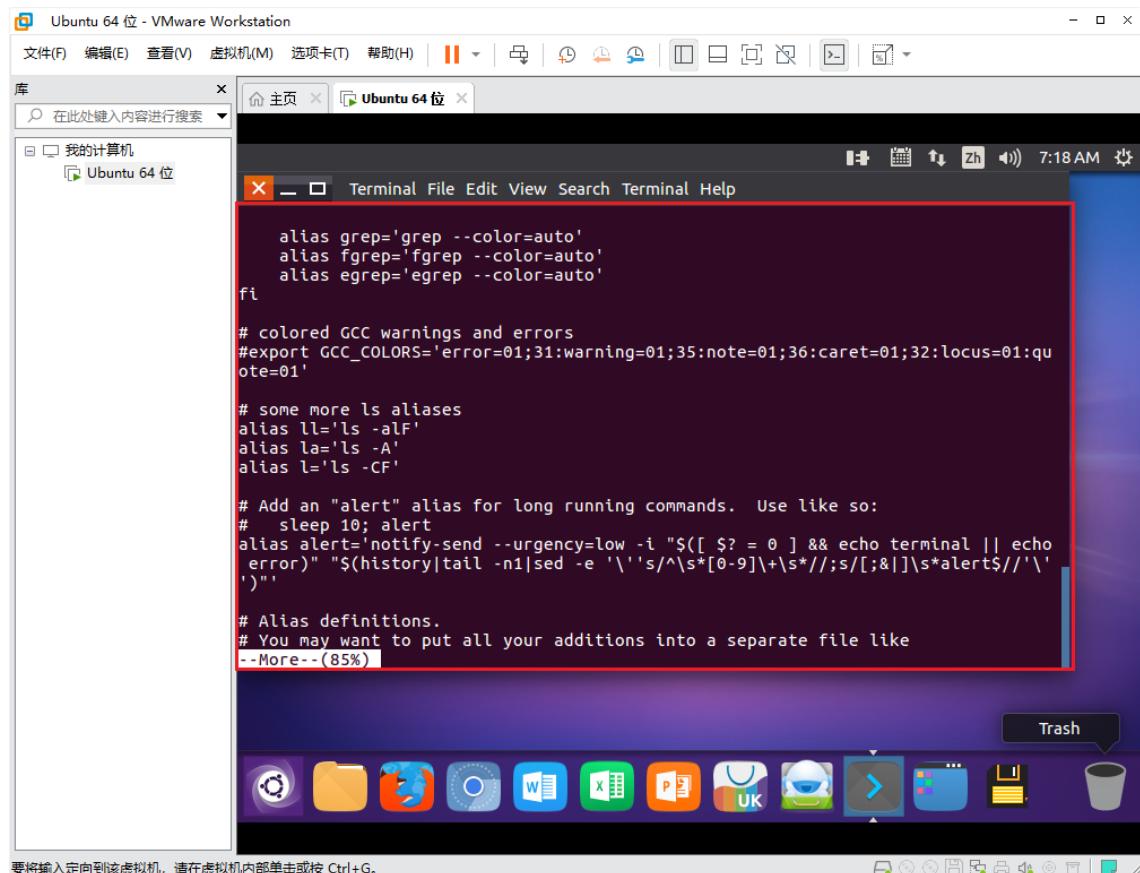
# set variable identifying the chroot you work in (used in the prompt below)
if [ -z "${debian_chroot:-}" ] && [ -r /etc/debian_chroot ]; then
    debian_chroot=$(cat /etc/debian_chroot)
fi

# set a fancy prompt (non-color, unless we know we "want" color)
case "$TERM" in
--More--(32%)
```

## 实验一

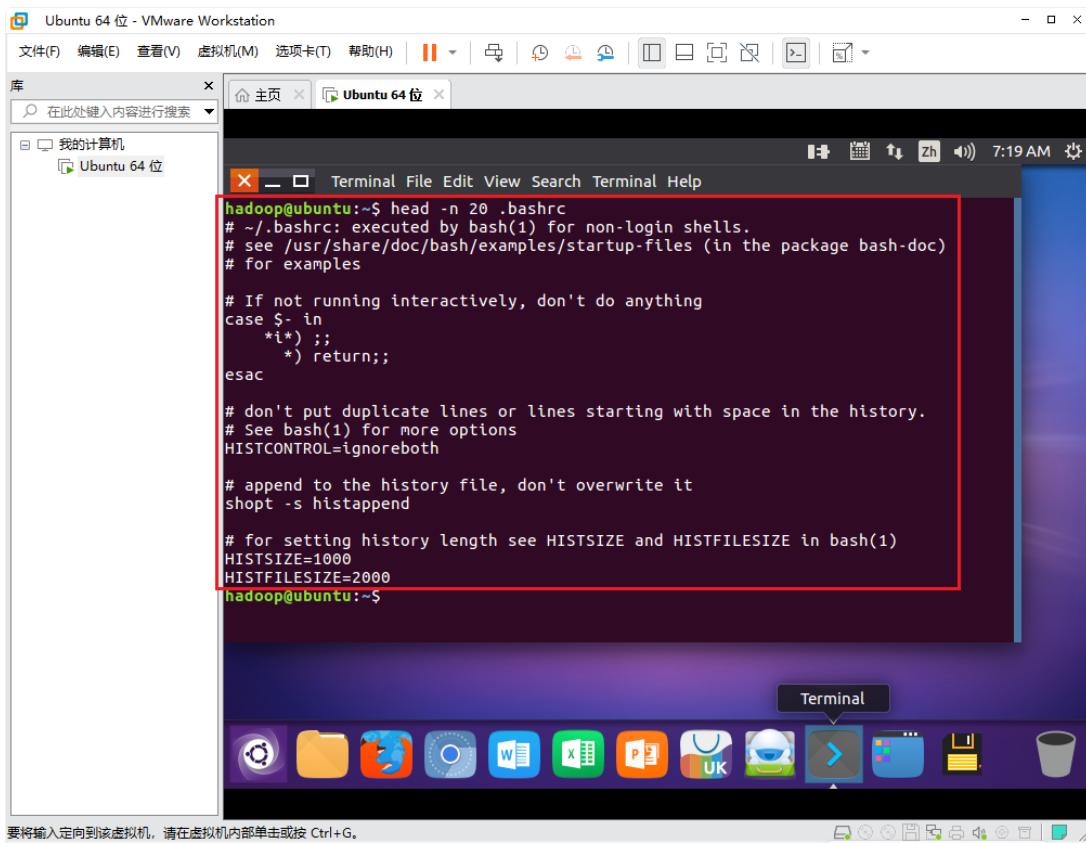


## 实验一



## 实验一

### (18) 查看用户主文件夹下的`.bashrc`文件内容的前 20 行



```
hadoop@ubuntu:~$ head -n 20 .bashrc
# ~/.bashrc: executed by bash(1) for non-login shells.
# see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
# for examples

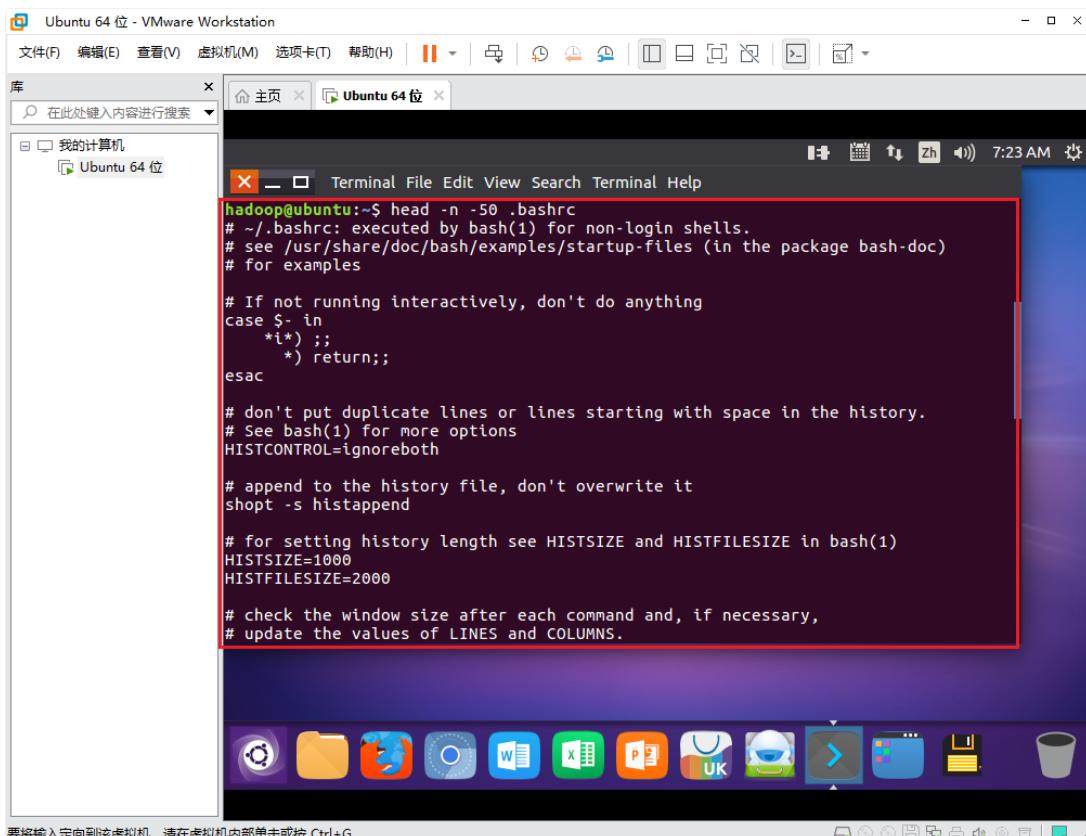
# If not running interactively, don't do anything
case $- in
    *i*) ;;
    *) return;;
esac

# don't put duplicate lines or lines starting with space in the history.
# See bash(1) for more options
HISTCONTROL=ignoreboth

# append to the history file, don't overwrite it
shopt -s histappend

# for setting history length see HISTSIZE and HISTFILESIZE in bash(1)
HISTSIZE=1000
HISTFILESIZE=2000
hadoop@ubuntu:~$
```

### (19) 查看主文件夹下.bashrc 文件内容，后面 50 行不显示，只显示前面几行



```
hadoop@ubuntu:~$ head -n -50 .bashrc
# ~/.bashrc: executed by bash(1) for non-login shells.
# see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
# for examples

# If not running interactively, don't do anything
case $- in
    *i*) ;;
    *) return;;
esac

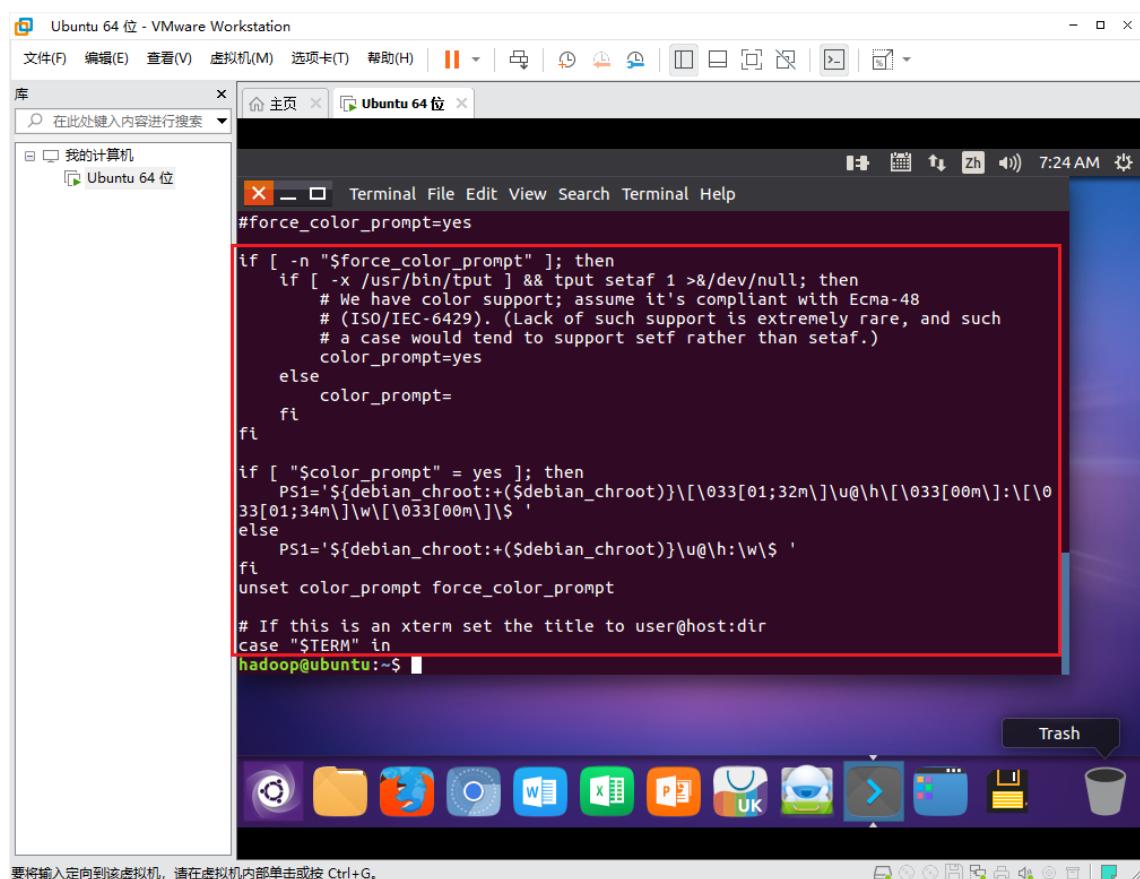
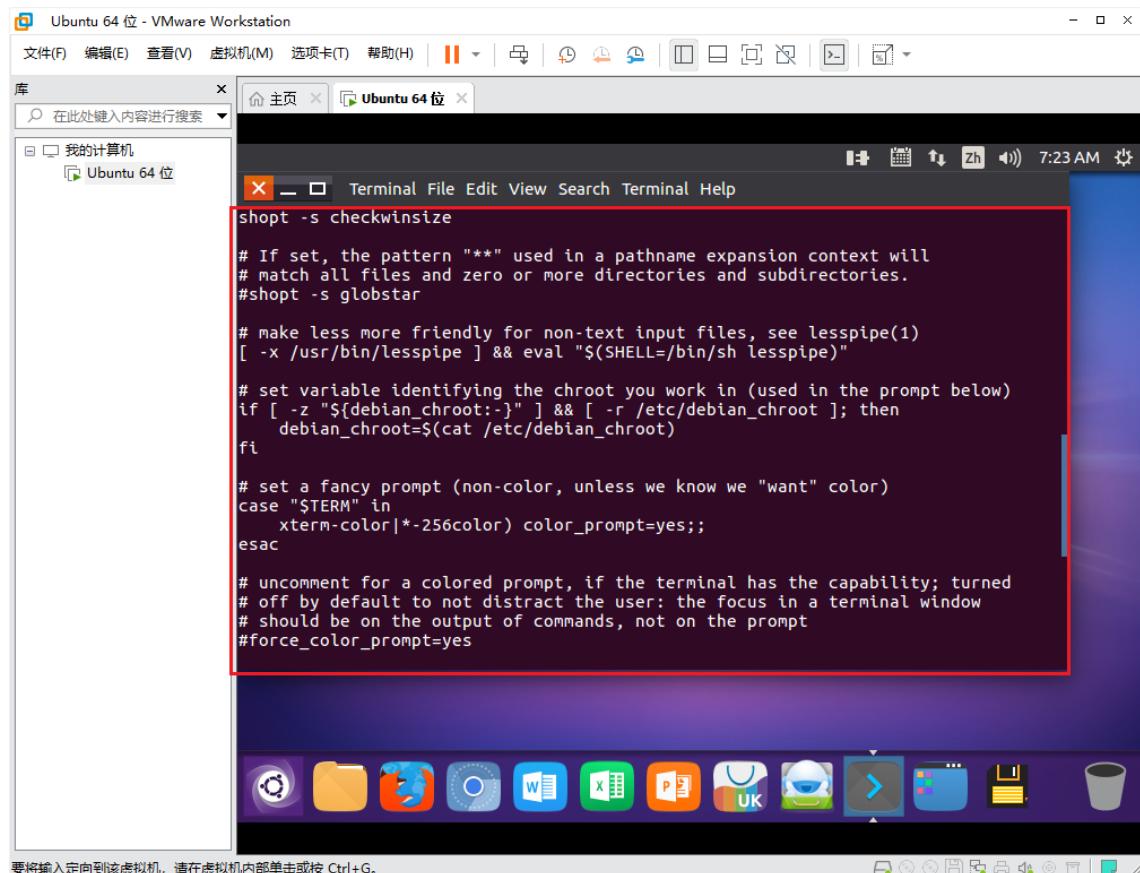
# don't put duplicate lines or lines starting with space in the history.
# See bash(1) for more options
HISTCONTROL=ignoreboth

# append to the history file, don't overwrite it
shopt -s histappend

# for setting history length see HISTSIZE and HISTFILESIZE in bash(1)
HISTSIZE=1000
HISTFILESIZE=2000

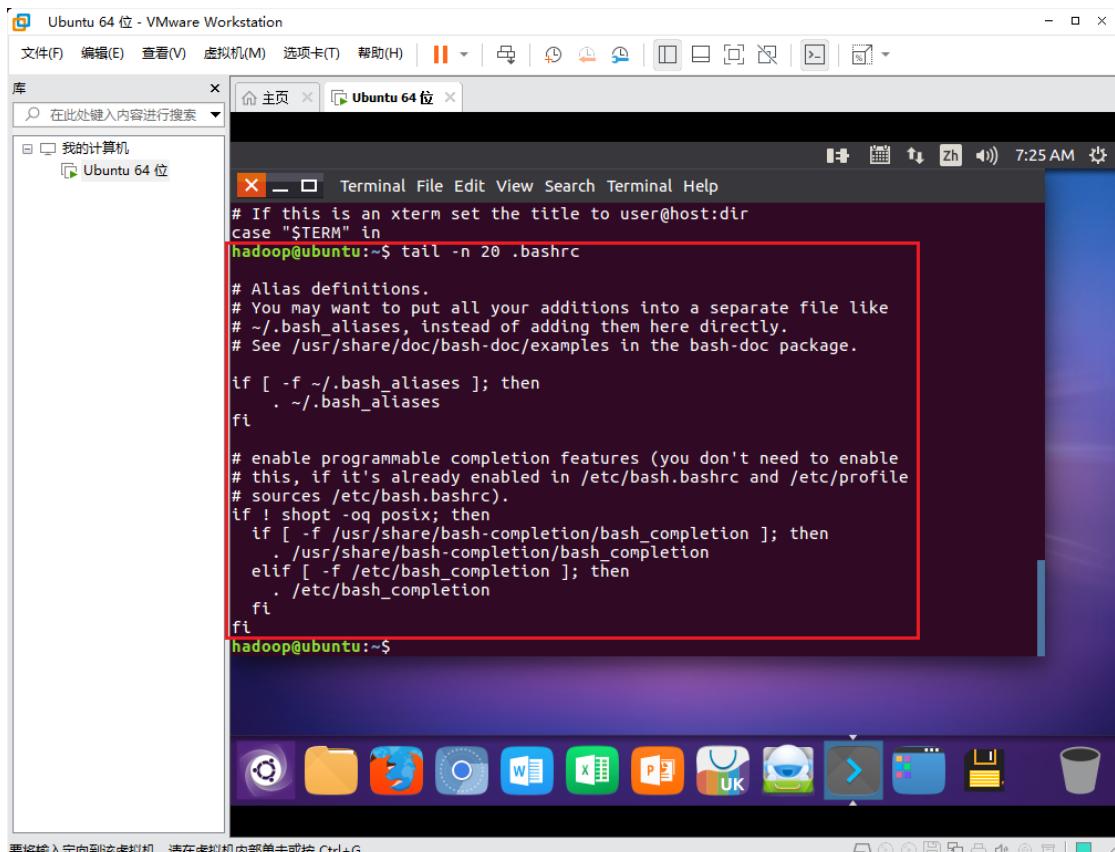
# check the window size after each command and, if necessary,
# update the values of LINES and COLUMNS.
```

## 实验一



## 实验一

### (20) 查看用户主文件夹下的`.bashrc`文件内容的最后 20 行



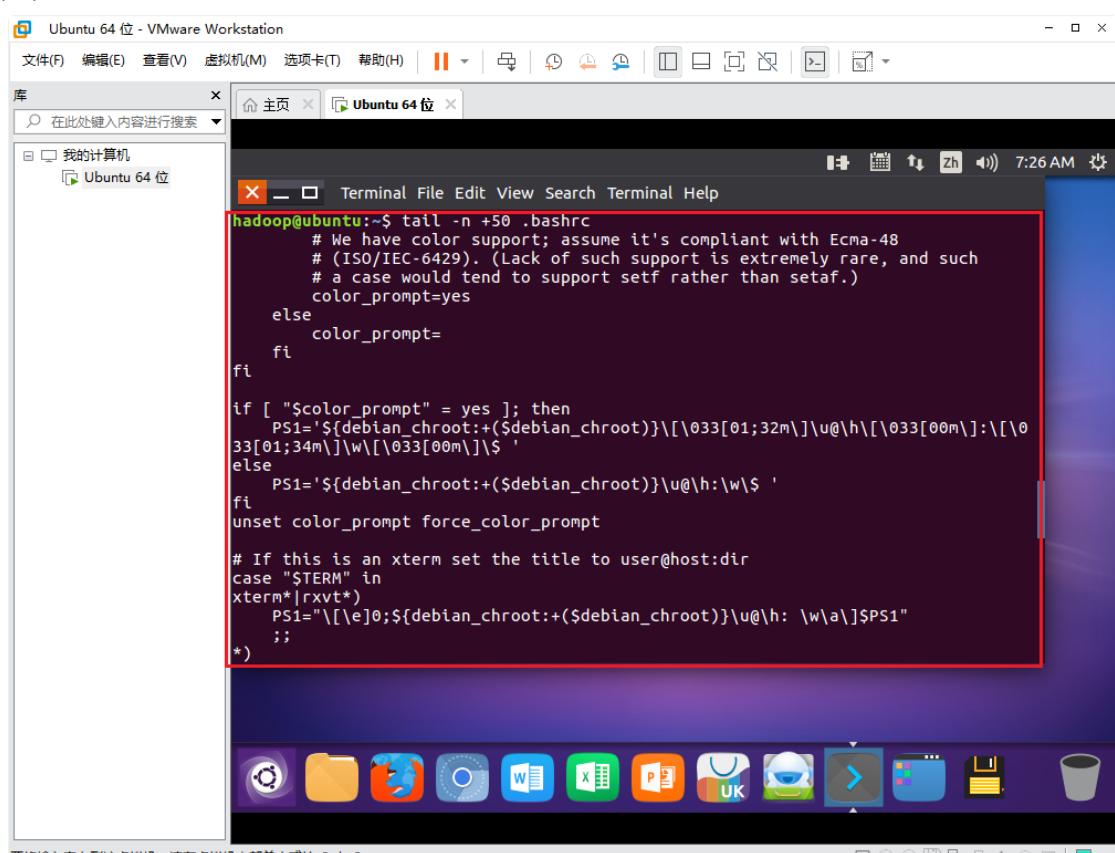
```
# If this is an xterm set the title to user@host:dir
case "$TERM" in
hadoop@ubuntu:~$ tail -n 20 .bashrc

# Alias definitions.
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
  . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi
hadoop@ubuntu:~$
```

### (21) 查看用户主文件夹下的`.bashrc`文件内容，只列出 50 行之后的数据

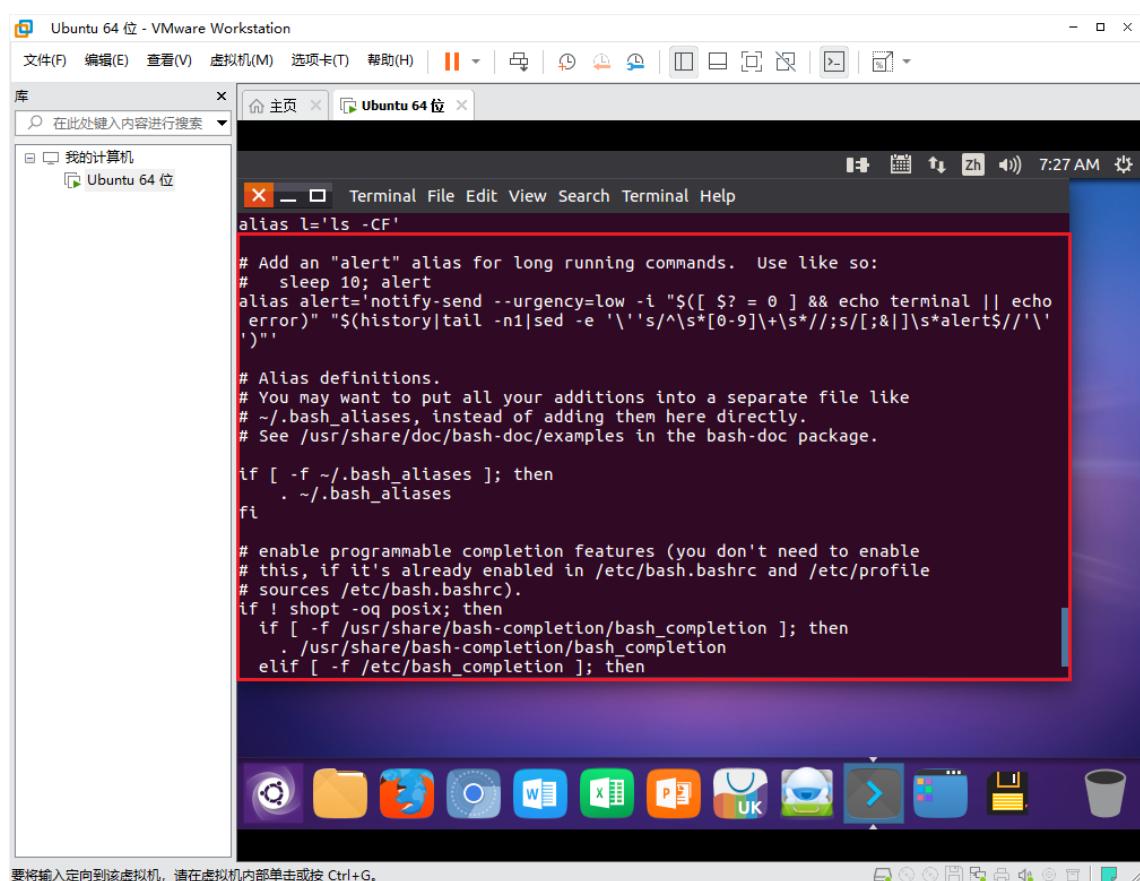
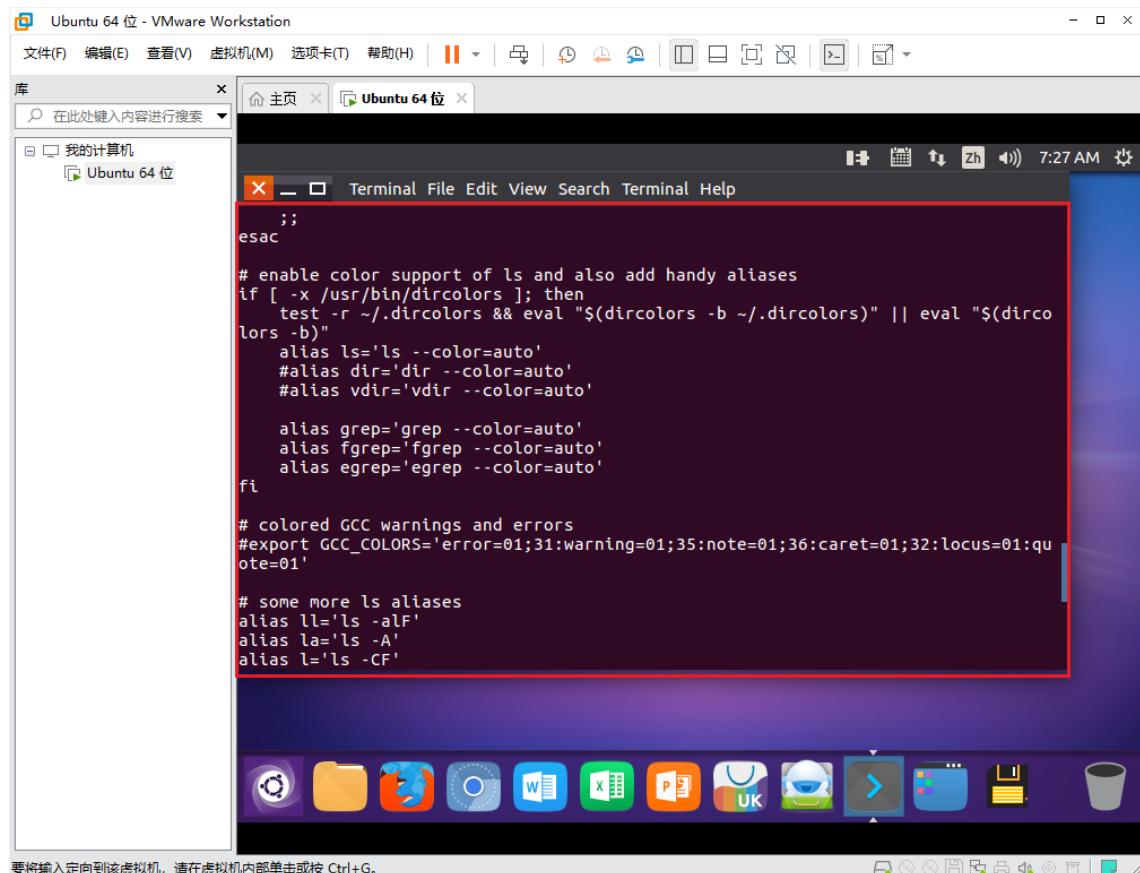


```
hadoop@ubuntu:~$ tail -n +50 .bashrc
# We have color support; assume it's compliant with Ecma-48
# (ISO/IEC-6429). (Lack of such support is extremely rare, and such
# a case would tend to support setf rather than setaf.)
color_prompt=yes
else
  color_prompt=
fi
fi

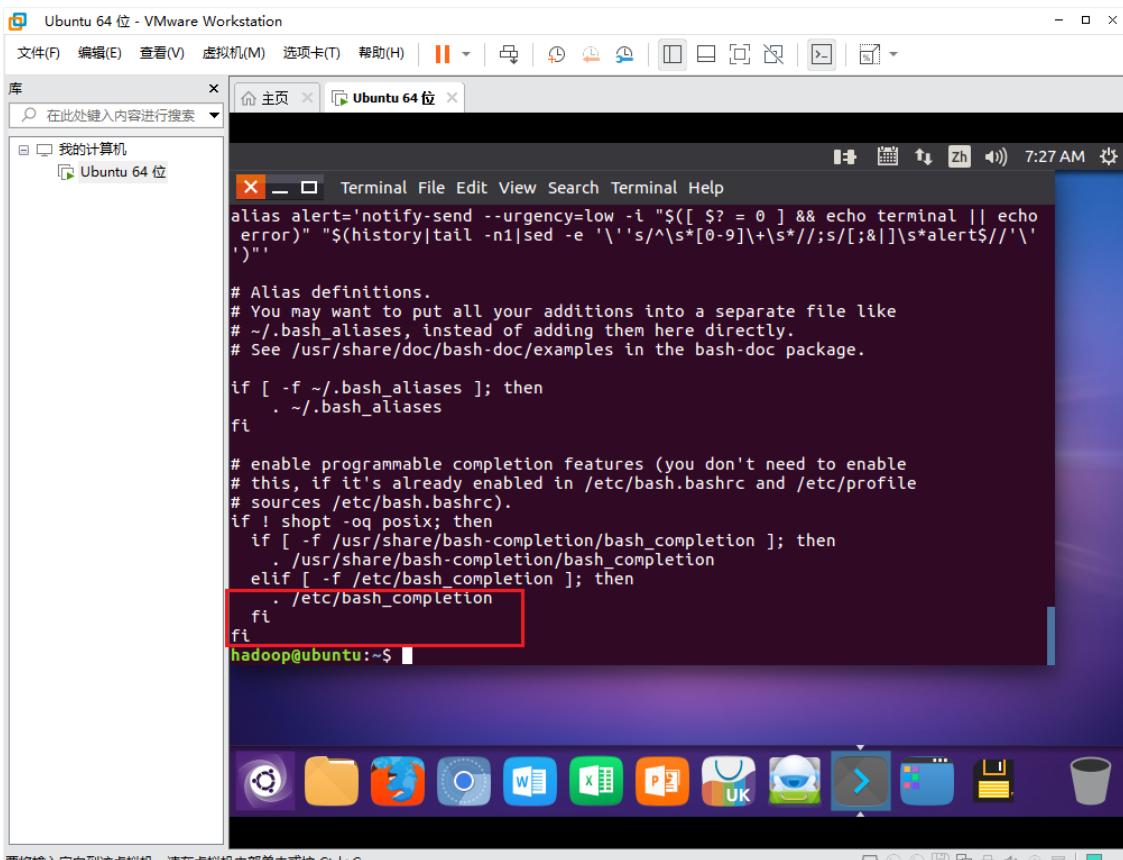
if [ "$color_prompt" = yes ]; then
  PS1='${debian_chroot:+($debian_chroot)}[\u033[01;32m]\u0@\h[\u033[00m]:[\u033[01;34m]\w[\u033[00m]\$ '
else
  PS1='${debian_chroot:+($debian_chroot)}\u0@\h:\w\$ '
fi
unset color_prompt force_color_prompt

# If this is an xterm set the title to user@host:dir
case "$TERM" in
xterm|rxvt*)
  PS1="\[\e[0;${debian_chroot:+($debian_chroot)}\u0@\h: \w\]${PS1}"
;;
*)
```

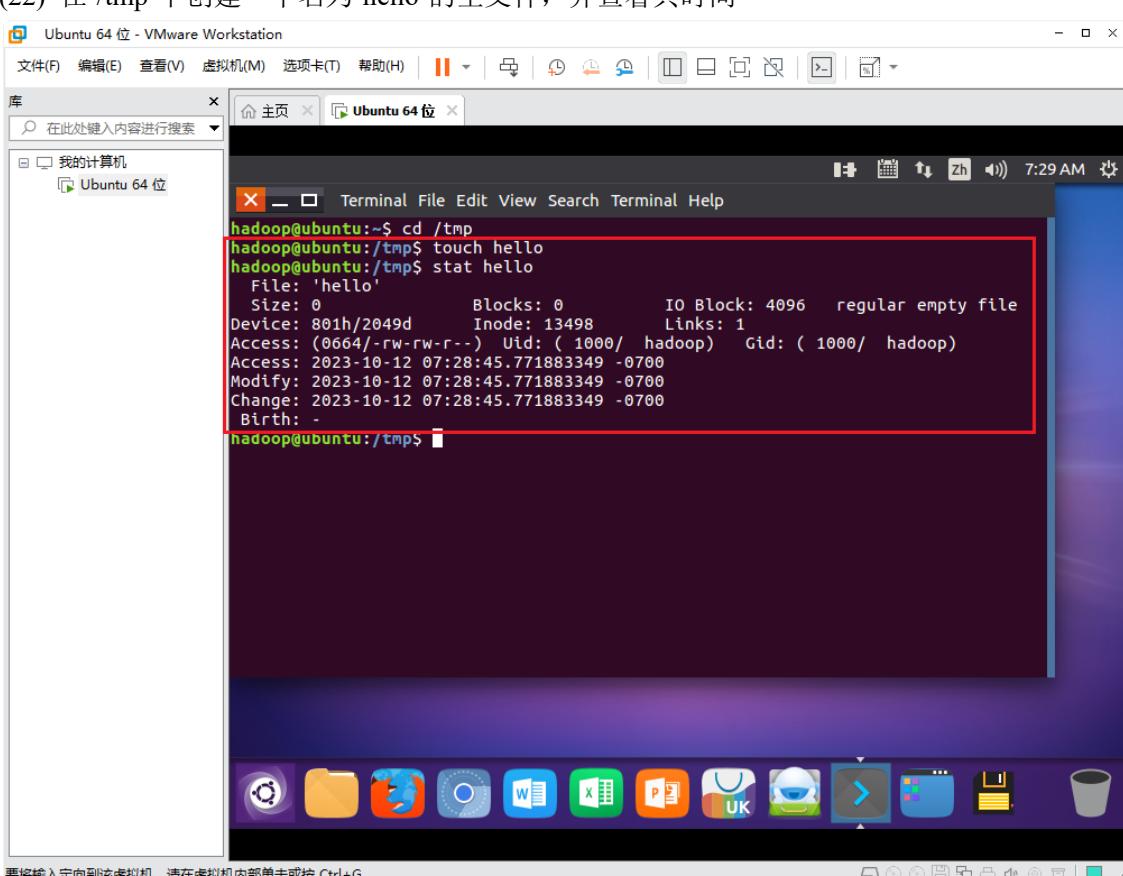
## 实验一



## 实验一



(22) 在`/tmp`下创建一个名为`hello`的空文件，并查看其时间



## 实验一

(23) 修改'hello'文件，将其日期调整为 5 天前

Ubuntu 64 位 - VMware Workstation

文件(F) 编辑(E) 查看(V) 虚拟机(M) 选项卡(T) 帮助(H) || |

库 在此处键入内容进行搜索

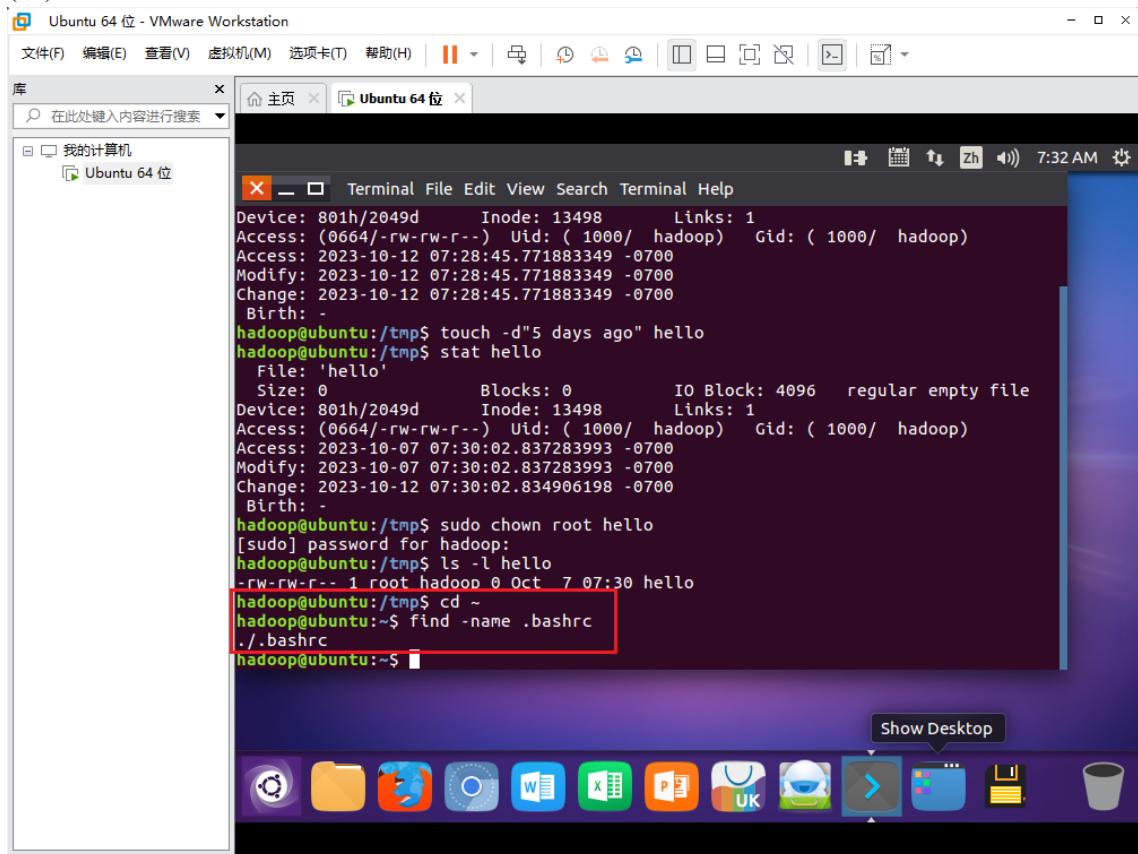
我的计算机 Ubuntu 64 位

Ubuntu 64 位

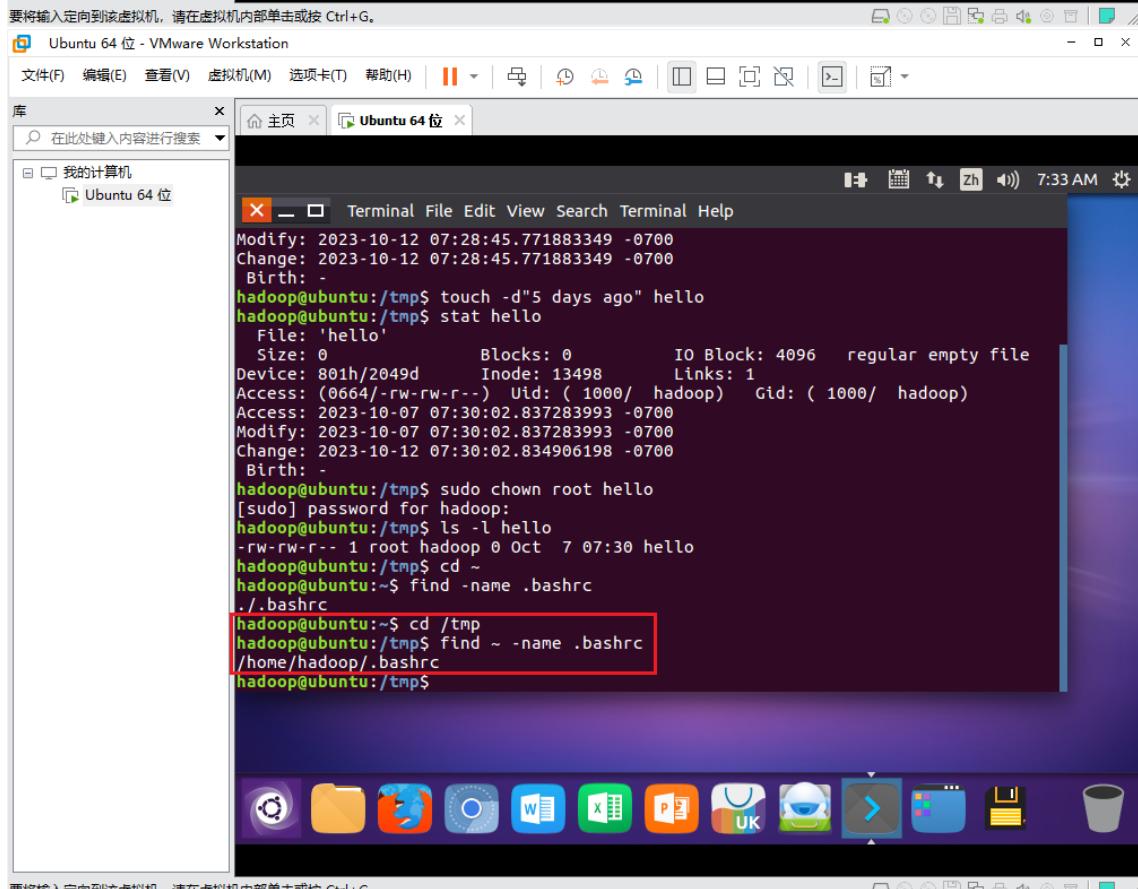
```
Terminal File Edit View Search Terminal Help
hadoop@ubuntu:~$ cd /tmp
hadoop@ubuntu:/tmp$ touch hello
hadoop@ubuntu:/tmp$ stat hello
  File: 'hello'
  Size: 0          Blocks: 0          IO Block: 4096   regular empty file
Device: 801h/2049d      Inode: 13498      Links: 1
Access: 0664/-rw-rw-r--  Uid: ( 1000/ hadoop)  Gid: ( 1000/ hadoop)
Access: 2023-10-12 07:28:45.771883349 -0700
Modify: 2023-10-12 07:28:45.771883349 -0700
Change: 2023-10-12 07:28:45.771883349 -0700
 Birth: -
hadoop@ubuntu:/tmp$ touch -d "5 days ago" hello
hadoop@ubuntu:/tmp$ stat hello
  File: 'hello'
  Size: 0          Blocks: 0          IO Block: 4096   regular empty file
Device: 801h/2049d      Inode: 13498      Links: 1
Access: 0664/-rw-rw-r--  Uid: ( 1000/ hadoop)  Gid: ( 1000/ hadoop)
Access: 2023-10-07 07:30:02.837283993 -0700
Modify: 2023-10-07 07:30:02.837283993 -0700
Change: 2023-10-12 07:30:02.834906198 -0700
 Birth: -
hadoop@ubuntu:/tmp$
```

(4) 将“11”文件的所有者更改为：“帐号”并查看其属性。

(25) 查找用户主文件夹下文件名为`.bashrc`的文件



```
Device: 801h/2049d Inode: 13498 Links: 1
Access: (0664/-rw-rw-r--)Uid: ( 1000/ hadoop) Gid: ( 1000/ hadoop)
Access: 2023-10-12 07:28:45.771883349 -0700
Modify: 2023-10-12 07:28:45.771883349 -0700
Change: 2023-10-12 07:28:45.771883349 -0700
Birth: -
hadoop@ubuntu:/tmp$ touch -d"5 days ago" hello
hadoop@ubuntu:/tmp$ stat hello
  File: 'hello'
  Size: 0          Blocks: 0          IO Block: 4096   regular empty file
Device: 801h/2049d Inode: 13498 Links: 1
Access: (0664/-rw-rw-r--)Uid: ( 1000/ hadoop) Gid: ( 1000/ hadoop)
Access: 2023-10-07 07:30:02.837283993 -0700
Modify: 2023-10-07 07:30:02.837283993 -0700
Change: 2023-10-12 07:30:02.834906198 -0700
Birth: -
hadoop@ubuntu:/tmp$ sudo chown root hello
[sudo] password for hadoop:
hadoop@ubuntu:/tmp$ ls -l hello
-rw-rw-r-- 1 root hadoop 0 Oct  7 07:30 hello
hadoop@ubuntu:/tmp$ cd ~
hadoop@ubuntu:~$ find -name .bashrc
./.bashrc
hadoop@ubuntu:~$
```



```
Modify: 2023-10-12 07:28:45.771883349 -0700
Change: 2023-10-12 07:28:45.771883349 -0700
Birth: -
hadoop@ubuntu:/tmp$ touch -d"5 days ago" hello
hadoop@ubuntu:/tmp$ stat hello
  File: 'hello'
  Size: 0          Blocks: 0          IO Block: 4096   regular empty file
Device: 801h/2049d Inode: 13498 Links: 1
Access: (0664/-rw-rw-r--)Uid: ( 1000/ hadoop) Gid: ( 1000/ hadoop)
Access: 2023-10-07 07:30:02.837283993 -0700
Modify: 2023-10-07 07:30:02.837283993 -0700
Change: 2023-10-12 07:30:02.834906198 -0700
Birth: -
hadoop@ubuntu:/tmp$ sudo chown root hello
[sudo] password for hadoop:
hadoop@ubuntu:/tmp$ ls -l hello
-rw-rw-r-- 1 root hadoop 0 Oct  7 07:30 hello
hadoop@ubuntu:/tmp$ cd ~
hadoop@ubuntu:~$ find -name .bashrc
./.bashrc
hadoop@ubuntu:~$ cd /tmp
hadoop@ubuntu:/tmp$ find ~ -name .bashrc
/home/hadoop/.bashrc
hadoop@ubuntu:/tmp$
```

## 实验一

(26) 在根目录`/`下创建文件夹`test`，然后打包成`test.tar.gz`

```
hadoop@ubuntu:/tmp$ stat hello
  File: 'hello'
  Size: 0          Blocks: 0          IO Block: 4096   regular empty file
Device: 801h/2049d  Inode: 13498      Links: 1
Access: (0664/-rw-rw-r--)  Uid: ( 1000/ hadoop)  Gid: ( 1000/ hadoop)
Access: 2023-10-07 07:30:02.837283993 -0700
Modify: 2023-10-07 07:30:02.837283993 -0700
Change: 2023-10-12 07:30:02.834906198 -0700
 Birth: -
hadoop@ubuntu:/tmp$ sudo chown root hello
[sudo] password for hadoop:
hadoop@ubuntu:/tmp$ ls -l hello
-rw-rw-r-- 1 root hadoop 0 Oct  7 07:30 hello
hadoop@ubuntu:/tmp$ cd ~
hadoop@ubuntu:~$ find -name .bashrc
./.bashrc
hadoop@ubuntu:~$ cd /tmp
hadoop@ubuntu:/tmp$ find ~ -name .bashrc
/hadoop/.bashrc
hadoop@ubuntu:/tmp$ cd /
hadoop@ubuntu:/$ sudo mkdir /test
hadoop@ubuntu:/$ sudo tar -zcv -f /test.tar.gz test
test/
hadoop@ubuntu:/$
```

```
Access: (0664/-rw-rw-r--)  Uid: ( 1000/ hadoop)  Gid: ( 1000/ hadoop)
Access: 2023-10-07 07:30:02.837283993 -0700
Modify: 2023-10-07 07:30:02.837283993 -0700
Change: 2023-10-12 07:30:02.834906198 -0700
 Birth: -
hadoop@ubuntu:/tmp$ sudo chown root hello
[sudo] password for hadoop:
hadoop@ubuntu:/tmp$ ls -l hello
-rw-rw-r-- 1 root hadoop 0 Oct  7 07:30 hello
hadoop@ubuntu:/tmp$ cd ~
hadoop@ubuntu:~$ find -name .bashrc
./.bashrc
hadoop@ubuntu:~$ cd /tmp
hadoop@ubuntu:/tmp$ find ~ -name .bashrc
/hadoop/.bashrc
hadoop@ubuntu:/tmp$ cd /
hadoop@ubuntu:/$ sudo mkdir /test
hadoop@ubuntu:/$ sudo tar -zcv -f /test.tar.gz test
test/
hadoop@ubuntu:/$ ls
bin  dev  initrd.img  lost+found  opt  run  srv  test.tar.gz  var
boot  etc  lib  media  proc  sbin  sys  vmlinuz
cdrom  home  lib64  mnt  root  snap  test  usr
hadoop@ubuntu:/$
```

## 实验一

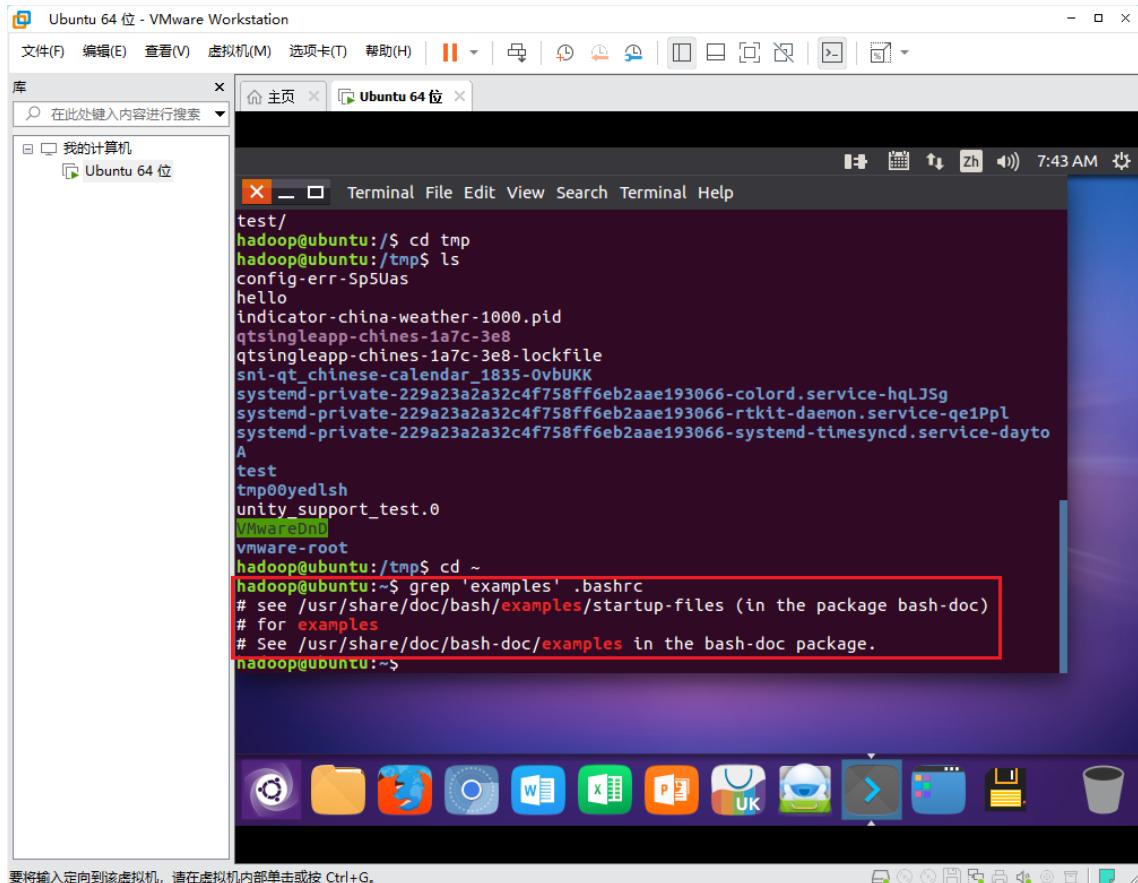
### (27) 解压缩到`/tmp`目录

```
Modify: 2023-10-07 07:30:02.837283993 -0700
Change: 2023-10-12 07:30:02.834906198 -0700
Birth: -
hadoop@ubuntu:/tmp$ sudo chown root hello
[sudo] password for hadoop:
hadoop@ubuntu:/tmp$ ls -l hello
-rw-rw-r-- 1 root hadoop 0 Oct  7 07:30 hello
hadoop@ubuntu:/tmp$ cd ~
hadoop@ubuntu:~$ find -name .bashrc
./.bashrc
hadoop@ubuntu:~$ cd /tmp
hadoop@ubuntu:/tmp$ find ~ -name .bashrc
/hadoop/.bashrc
hadoop@ubuntu:/tmp$ cd /
hadoop@ubuntu:/$ sudo mkdir /test
hadoop@ubuntu:/$ sudo tar -zcv -f /test.tar.gz test
test/
hadoop@ubuntu:/$ ls
bin  dev  initrd.img  lost+found  opt  run  srv  test.tar.gz  var
boot  etc  lib  media  proc  sbin  sys  tmp  vmlinuz
cdrom  home  lib64  mnt  root  snap  test  usr
hadoop@ubuntu:/$ sudo tar -zcv -f /test.tar.gz test -C /tmp
test/
hadoop@ubuntu:/$
```

```
hadoop@ubuntu:/$ ls
bin  dev  initrd.img  lost+found  opt  run  srv  test.tar.gz  var
boot  etc  lib  media  proc  sbin  sys  tmp  vmlinuz
cdrom  home  lib64  mnt  root  snap  test  usr
hadoop@ubuntu:/$ sudo tar -zcv -f /test.tar.gz test -C /tmp
test/
hadoop@ubuntu:/$ cd tmp
hadoop@ubuntu:/tmp$ ls
config-err-Sp5Usas
hello
indicator-china-weather-1000.pid
qtsingleapp-chines-1a7c-3e8
qtsingleapp-chines-1a7c-3e8-lockfile
sni-qt_chinese-calendar_1835-0vbUKK
systemd-private-229a23a2a32c4f758ff6eb2aae193066-colord.service-hqLJSg
systemd-private-229a23a2a32c4f758ff6eb2aae193066-rtkit-daemon.service-qe1Ppl
systemd-private-229a23a2a32c4f758ff6eb2aae193066-systemd-timesyncd.service-dayto
A
test
tmp00yedlsh
unity_support_test.0
VMwareBnD
vmware-root
hadoop@ubuntu:/tmp$
```

(28) 从`~/.bashrc`文件中查找字符串'examples'



(29) 在`~/.bashrc`中设置 Java 环境变量

```
export JAVA_HOME=/usr/lib/jvm/jdk1.8.0_162
export JRE_HOME=${JAVA_HOME}/jre
export CLASSPATH=.:${JAVA_HOME}/lib:${JRE_HOME}/lib
export PATH=${JAVA_HOME}/bin:$PATH
# ~/.bashrc: executed by bash(1) for non-login shells.
# see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
# for examples

# If not running interactively, don't do anything
case $- in
    *i*) ;;
    *) return;;
esac

# don't put duplicate lines or lines starting with space in the history.
# See bash(1) for more options
HISTCONTROL=ignoreboth

# append to the history file, don't overwrite it
shopt -s histappend

# for setting history length see HISTSIZE and HISTFILESIZE in bash(1)
HISTSIZE=1000
HISTFILESIZE=2000

# check the window size after each command and, if necessary,
# update the values of LINES and COLUMNS.
shopt -s checkwinsize

# If set, the pattern "*" used in a pathname expansion context will
# match all files and zero or more directories and subdirectories.
#shopt -s globstar

# make less more friendly for non-text input files, see lesspipe(1)
[ -x /usr/bin/lesspipe ] && eval "$(SHELL=/bin/sh lesspipe)"

# set variable identifying the chroot you work in (used in the prompt below)
if [ -z "${debian_chroot:-}" ] && [ -r /etc/debian_chroot ]; then
    debian_chroot=$(cat /etc/debian_chroot)
fi

# set a fancy prompt (non-color, unless we know we "want" color)
case "$TERM" in
    xterm-color|*256color) color_prompt=yes;;
esac
"~/.bashrc" 121L, 3934C
```

(30) 查看`JAVA\_HOME`变量的值

```
hadoop@ubuntu:~$ echo $JAVA_HOME  
/usr/lib/jvm/jdk1.8.0_162
```

(31)启动 Hadoop，并在 Hadoop 分布式文件系统（HDFS）中创建用户目录（假设 Hadoop 已经安装在`/usr/local/hadoop`目录下）

```
4670 ResourceManager  
hadoop@ubuntu:/usr/local/hadoop$ ./bin/hdfs dfs -mkdir -p /user/hadoop  
hadoop@ubuntu:/usr/local/hadoop$ █
```

(32)在该用户目录下创建一个名为`text`的文件夹，并查看 HDFS 中的文件列表

```
hadoop@ubuntu:/usr/local/hadoop$ ./bin/hdfs dfs -mkdir -p /user/hadoop  
hadoop@ubuntu:/usr/local/hadoop$ ./bin/hdfs dfs -mkdir test  
hadoop@ubuntu:/usr/local/hadoop$ ./bin/hdfs dfs -put ~/.bashrc test  
2023-10-11 22:35:33,885 INFO sasl.SaslDataTransferClient: SASL encryption trust  
check: localHostTrusted = false, remoteHostTrusted = false  
hadoop@ubuntu:/usr/local/hadoop$ ./bin/hdfs dfs -ls test  
Found 1 items  
-rw-r--r-- 1 hadoop supergroup 3934 2023-10-11 22:35 test/.bashrc  
hadoop@ubuntu:/usr/local/hadoop$ █
```

(33)将用户主文件夹下的`.bashrc`文件上传到 HDFS 的`test`文件夹，并查看`test`文件夹中的内容

```
hadoop@ubuntu:/usr/local/hadoop$ ./bin/hdfs dfs -mkdir -p /user/hadoop  
hadoop@ubuntu:/usr/local/hadoop$ ./bin/hdfs dfs -mkdir test  
hadoop@ubuntu:/usr/local/hadoop$ ./bin/hdfs dfs -put ~/.bashrc test  
2023-10-11 22:35:33,885 INFO sasl.SaslDataTransferClient: SASL encryption trust  
check: localHostTrusted = false, remoteHostTrusted = false  
hadoop@ubuntu:/usr/local/hadoop$ ./bin/hdfs dfs -ls test  
Found 1 items  
-rw-r--r-- 1 hadoop supergroup 3934 2023-10-11 22:35 test/.bashrc  
hadoop@ubuntu:/usr/local/hadoop$ █
```

(34)将 HDFS 中的`test`文件夹拷贝到本机

```
put: test/.bashrc : file exists  
hadoop@ubuntu:/usr/local/hadoop$ ./bin/hdfs dfs -get ./test /home/hadoop/workspace  
2023-10-11 22:48:37,013 INFO sasl.SaslDataTransferClient: SASL encryption trust  
check: localHostTrusted = false, remoteHostTrusted = false  
hadoop@ubuntu:/usr/local/hadoop$ █
```

## 五、 实验结果

### (一) 熟悉常用的 Linux 操作

在这个实验中,我们学习了各种常用的Linux基本命令。这些命令包括cd、ls、mkdir、rmdir、cp、mv、rm、cat、tac、more、tail、touch、chown、find、tar、grep等,具体如下:

1. 使用 cd 命令, 我们学会了如何切换目录, 返回上一级目录以及返回用户主文件夹;
2. 使用 ls 命令, 我们学会了如何查看目录下的文件和子目录;
3. 使用 mkdir 命令, 我们学会了如何创建新目录, 并查看有多少目录存在;
4. 使用 rmdir 命令, 我们学会了如何删除空的目录;
5. 使用 cp 命令, 我们学会了如何复制文件或目录;
6. 使用 mv 命令, 我们学会了如何移动文件和目录, 以及如何更改文件名;
7. 使用 rm 命令, 我们学会了如何删除文件和目录;
8. 使用 cat、tac、more、tail 命令, 我们学会了如何查看文件的内容, 反向查看文件内容, 以及如何逐页查看文件内容;
9. 使用 touch 命令, 我们学会了如何创建新文件和修改文件的时间戳;
10. 使用 chown 命令, 我们学会了如何更改文件的所有者权限;
11. 使用 find 命令, 我们学会了如何查找文件;
12. 使用 tar 命令, 我们学会了如何打包和解压文件夹;
13. 使用 grep 命令, 我们学会了如何在文件中查找字符串。

### (二) 熟悉常用的 Hadoop 操作

在这个实验中,我们学习了一些常用的 Hadoop 操作。这些命令包括在 HDFS 中创建用户目录、创建文件夹、上传文件到 HDFS、查看 HDFS 中的文件列表以及将 HDFS 中的文件拷贝到本机, 具体任务如下:

1. 启动 Hadoop, 并在 HDFS 中创建用户目录;
2. 在用户目录下创建一个名为 text 的文件夹, 并查看 HDFS 中的文件列表;
3. 将用户主文件夹下的.bashrc 文件上传到 HDFS 的 test 文件夹, 并查看 test 文件夹中的内容;
4. 将 HDFS 中的 test 文件夹拷贝到本机。

总的而言, 所有实验任务均完成并达到预期。

本次实验我们学习者建立了对 Linux 基本命令和 Hadoop 操作的基础知识, 这些知识在日常的系统管理和大数据处理中都非常有用。通过这些实验, 我们获得了对 Linux 和 Hadoop 的实际操作经验, 为更高级的任务和项目奠定了坚实的基础。

## 六、 问题和解决

### 问题 1：

Ubuntu 22.04.2 LTS 中，FTP 并不会默认开启，需要手动开启才能使用 FileZilla 与主机通信以传输文件。

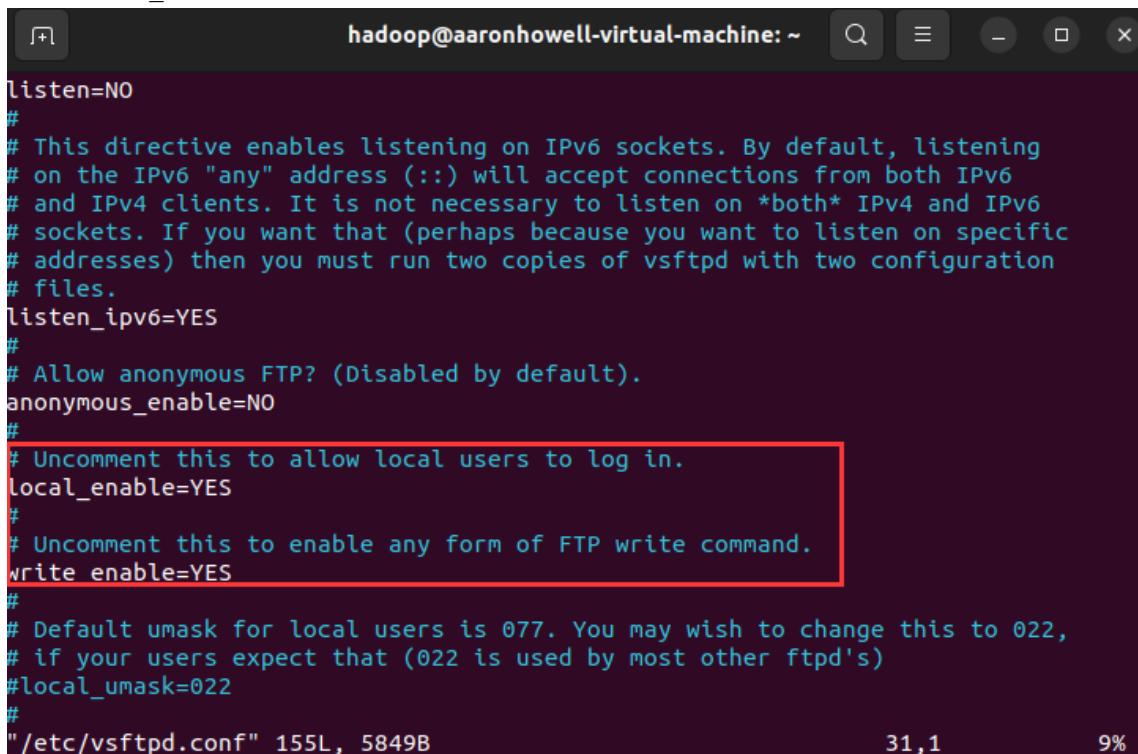
解决方案：

首先需要输入以下指令：

- sudo apt-get install vsftpd
- sudo vi /etc/vsftpd.conf

打开 vsftpd.conf，进行修改：

- local\_enable=YES
- write\_enable=YES



```
hadoop@aaronhowell-virtual-machine: ~
listen=NO
#
# This directive enables listening on IPv6 sockets. By default, listening
# on the IPv6 "any" address (::) will accept connections from both IPv6
# and IPv4 clients. It is not necessary to listen on *both* IPv4 and IPv6
# sockets. If you want that (perhaps because you want to listen on specific
# addresses) then you must run two copies of vsftpd with two configuration
# files.
listen_ipv6=YES
#
# Allow anonymous FTP? (Disabled by default).
anonymous_enable=NO
#
# Uncomment this to allow local users to log in.
local_enable=YES
#
# Uncomment this to enable any form of FTP write command.
write_enable=YES
#
# Default umask for local users is 077. You may wish to change this to 022,
# if your users expect that (022 is used by most other ftpt's)
#local_umask=022
#
"/etc/vsftpd.conf" 155L, 5849B          31,1          9%
```

在保存退出后，执行以下指令：

- sudo /etc/init.d/vsftpd restart

完成以上操作即可开启 FTP 功能。

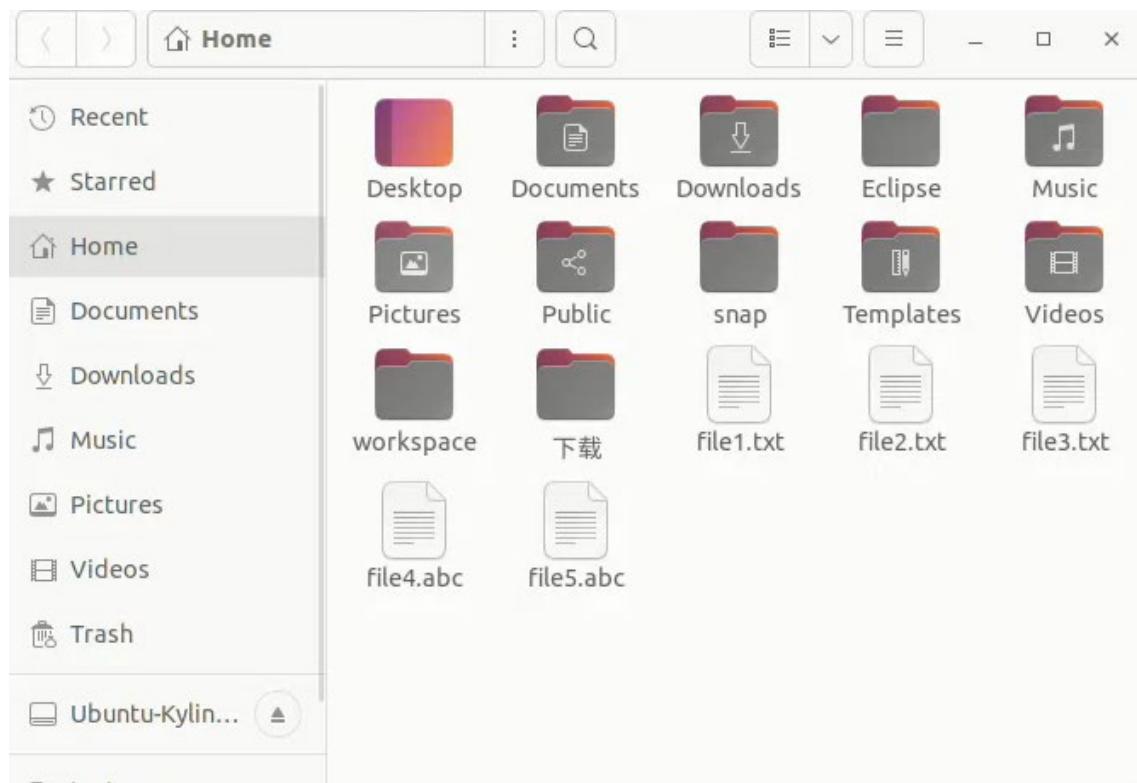
### 问题 2：

在 Ubuntu 22.04.2 LTS 中，目录可能默认中文，在使用 FileZilla 的时候会导致乱码。

解决方案：

需要将目录重命名为英文，可以使用以下指令：

- export LANG=en\_US
- xdg-user-dirs-gtk-update



完成以上操作即可将路径修改为英文，避免出现乱码问题。

#### 问题 3：

在输入命令和路径时，拼写错误可能会导致找不到文件或目录。确保仔细检查命令和路径，尤其是 Linux 对大小写敏感。

#### 解决方案：

仔细检查拼写，确保命令和路径正确。

#### 问题 4：

如果试图执行需要特殊权限的操作，例如在系统目录中创建或删除文件，但没有足够的权限，会收到错误消息。

#### 解决方案：

使用 `sudo` 命令以超级用户的身份执行操作，前提是您有相应的权限。但要小心，不要在系统关键目录中随意使用 `sudo` 命令，以免损坏系统。

#### 问题 5：

尝试进入、复制、移动或删除不存在的目录或文件，会出现错误。

#### 解决方案：

确保目录或文件存在，或者检查路径是否正确。

#### 问题 6：

在某些情况下，文件可能被其他进程锁定，阻止你对其进行操作。

解决方案：

等待其他进程完成对文件的操作，或者终止占用文件的进程。

问题 7：

如果尝试创建已存在的文件或目录，会收到错误。

解决方案：

检查文件或目录是否已存在，或者使用不同的名称。

问题 8：

在 Hadoop 实验中，可能会遇到与 Hadoop 配置和启动相关的问题，包括配置文件错误、端口冲突等。

解决方案：

仔细检查 Hadoop 配置，确保端口没有冲突，查看 Hadoop 的日志文件以获取更多信息，可以查找相应的解决方案或者调整配置。