

# 实验三-3

# 实验四-1

2014.5.17

# 本节课内容

- 1.实验三自己做实验，注意实验三不再有当堂检查，所以有git使用上的问题随时提出
- 2.实验四进行讲解

# 实验四 下载ppt和今天上课所需源代码

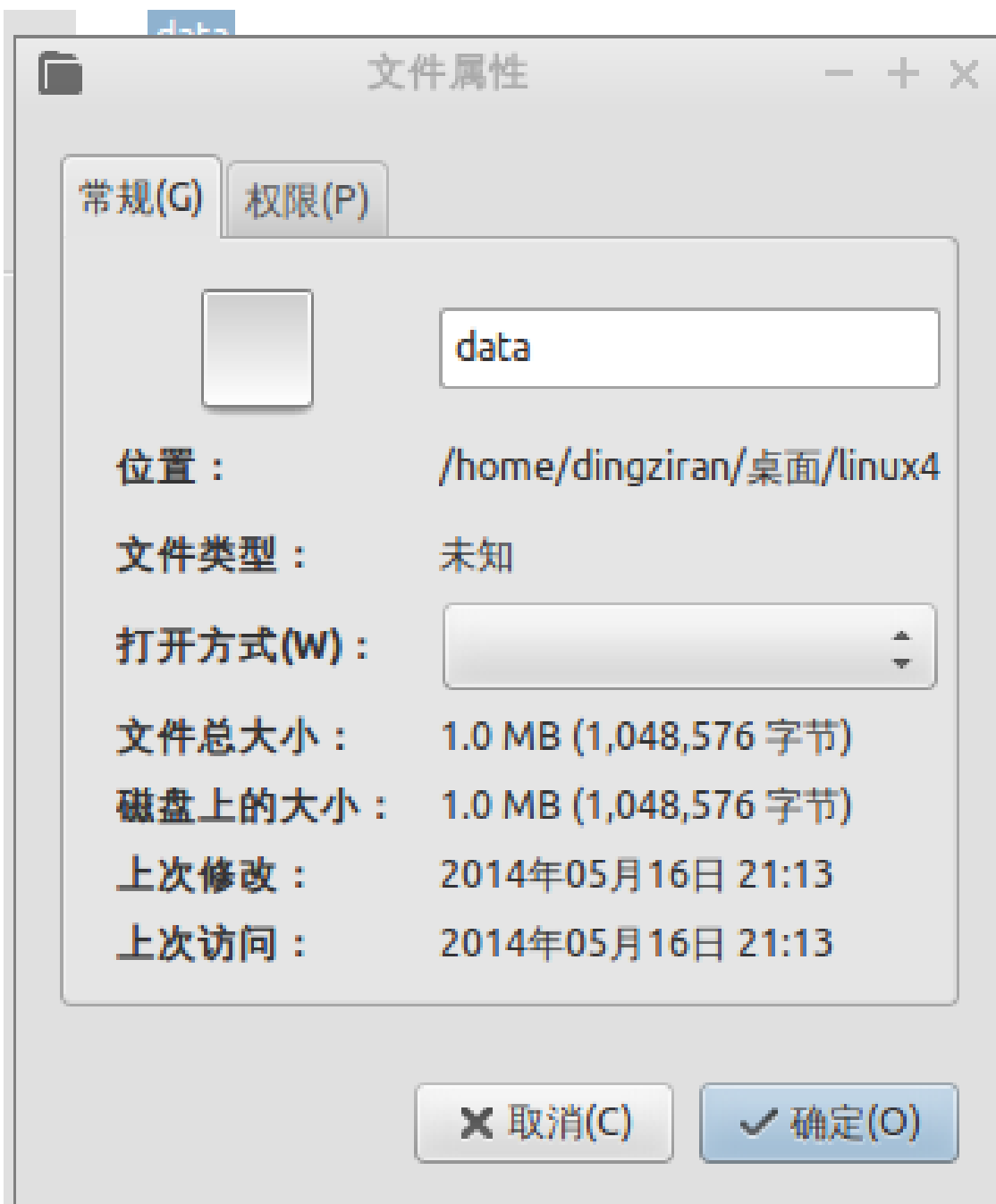
- 1.进入系统后，打开终端，安装git
- 2.在终端输入命令 `sudo apt-get install git`，然后输入系统密码
- 3.打开控制台，在想要存放工程的目录下，  
输入`git clone https://github.com/dingziranrr/linux4`
- 4.使用`cd linux4`进入工程目录
- 注意：之后就在这个目录进行实验，如果需要移动此目录，将此目录整个打包进行移动，最后提交的时候将此目录打包发给助教。

# 关于实验四所需要用的虚拟优盘

- 不需要使用真实优盘，按照如下操作可以制作虚拟优盘
- 1.使用dd命令创建一个自定义大小的空文件
- 2.使用mkfs.msdos命令将此文件格式化为fat16
- 3.使用mount命令将此文件挂载成优盘

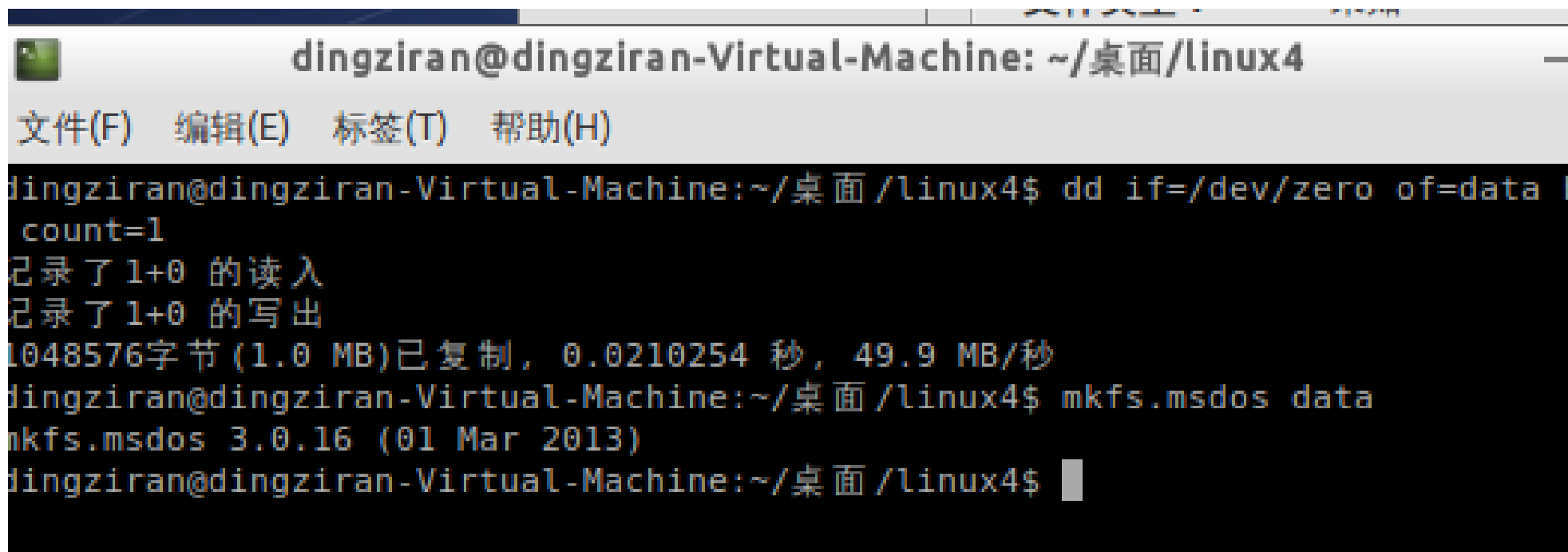
# 1.创建空文件

- 1 创建特定大小的文件最简单的方法是使用dd命令，dd命令会克隆给定的输入内容，然后将一份一模一样的副本写道输出
- 2 stdin，设备文件，普通文件等都可以作为输入， stdout，设备文件，普通文件等也可以作为输出
- 3 例子： `dd if=/dev/zero of=data bs=1M count=1`
- 该命令会在当前目录创建一个1MB大小的文件data，来看一下参数，if代表输入文件，of代表输出文件，bs代表以字节为单位的块大小，count表示要被复制的块数
- 4 我们将bs指定为1MB，count为1，于是得到了一个大小为1MB的文件，如果把count指定为2则得到2MB的文件
- 5 /dev/zero是一个字符设备，他会不断返回0值字节（\0）



## 2.格式化

- 使用msfs.msdos data命令

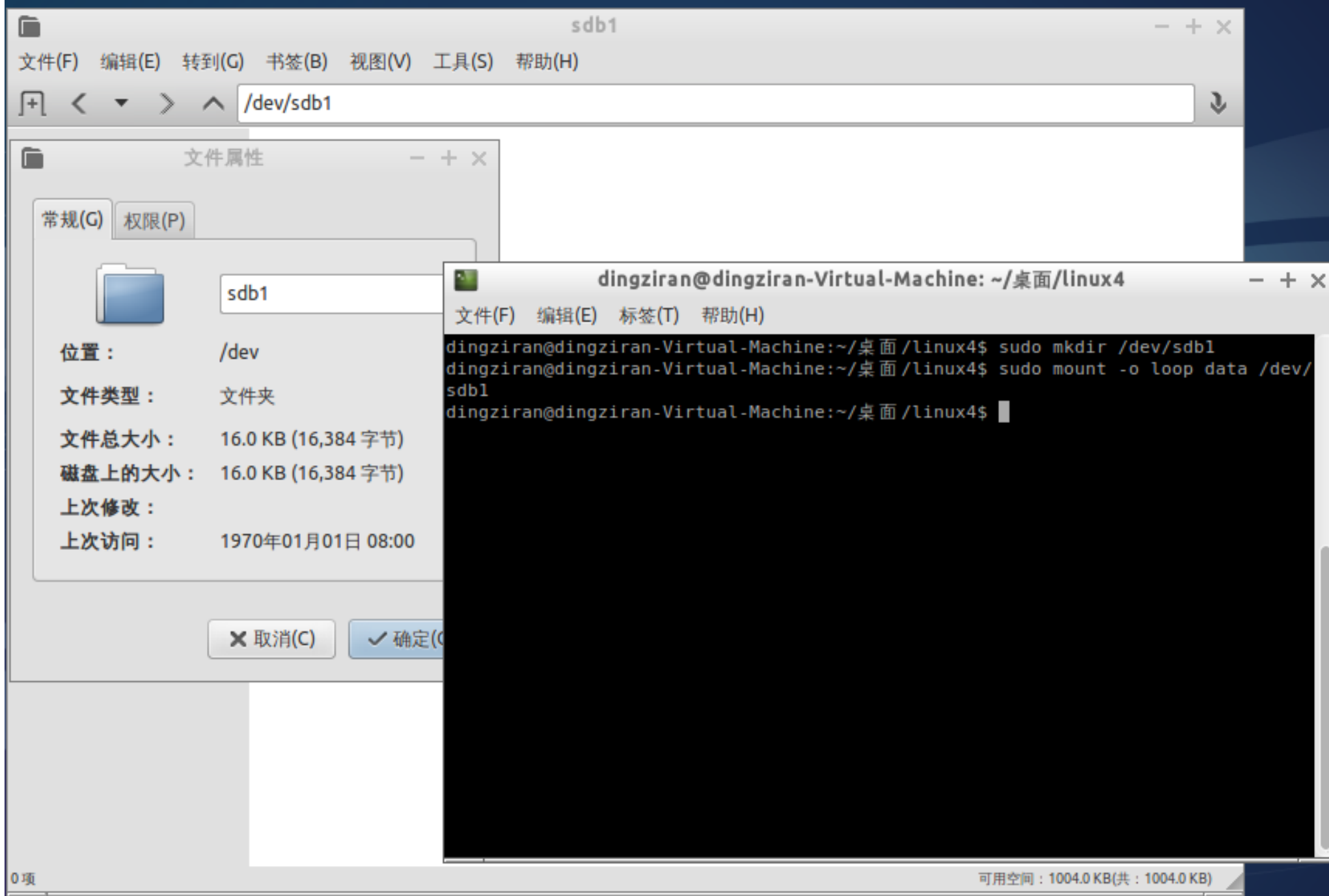
A terminal window titled 'dingziran@dingziran-Virtual-Machine: ~/桌面/linux4'. The window has a menu bar with '文件(F)', '编辑(E)', '标签(T)', and '帮助(H)'. The terminal shows the execution of 'dd if=/dev/zero of=data bs=1M count=1', which formats a 1MB file named 'data'. It reports '1048576字节 (1.0 MB)已复制, 0.0210254 秒, 49.9 MB/秒'. Following this, the 'mkfs.msdos data' command is run, displaying 'mkfs.msdos 3.0.16 (01 Mar 2013)'. The prompt returns to 'dingziran@dingziran-Virtual-Machine:~/桌面/linux4\$' with a cursor.

```
dingziran@dingziran-Virtual-Machine: ~/桌面/linux4
文件(F) 编辑(E) 标签(T) 帮助(H)
dingziran@dingziran-Virtual-Machine:~/桌面/linux4$ dd if=/dev/zero of=data bs=1M count=1
记录了1+0 的读入
记录了1+0 的写出
1048576字节 (1.0 MB)已复制, 0.0210254 秒, 49.9 MB/秒
dingziran@dingziran-Virtual-Machine:~/桌面/linux4$ mkfs.msdos data
mkfs.msdos 3.0.16 (01 Mar 2013)
dingziran@dingziran-Virtual-Machine:~/桌面/linux4$
```

### 3.挂载优盘

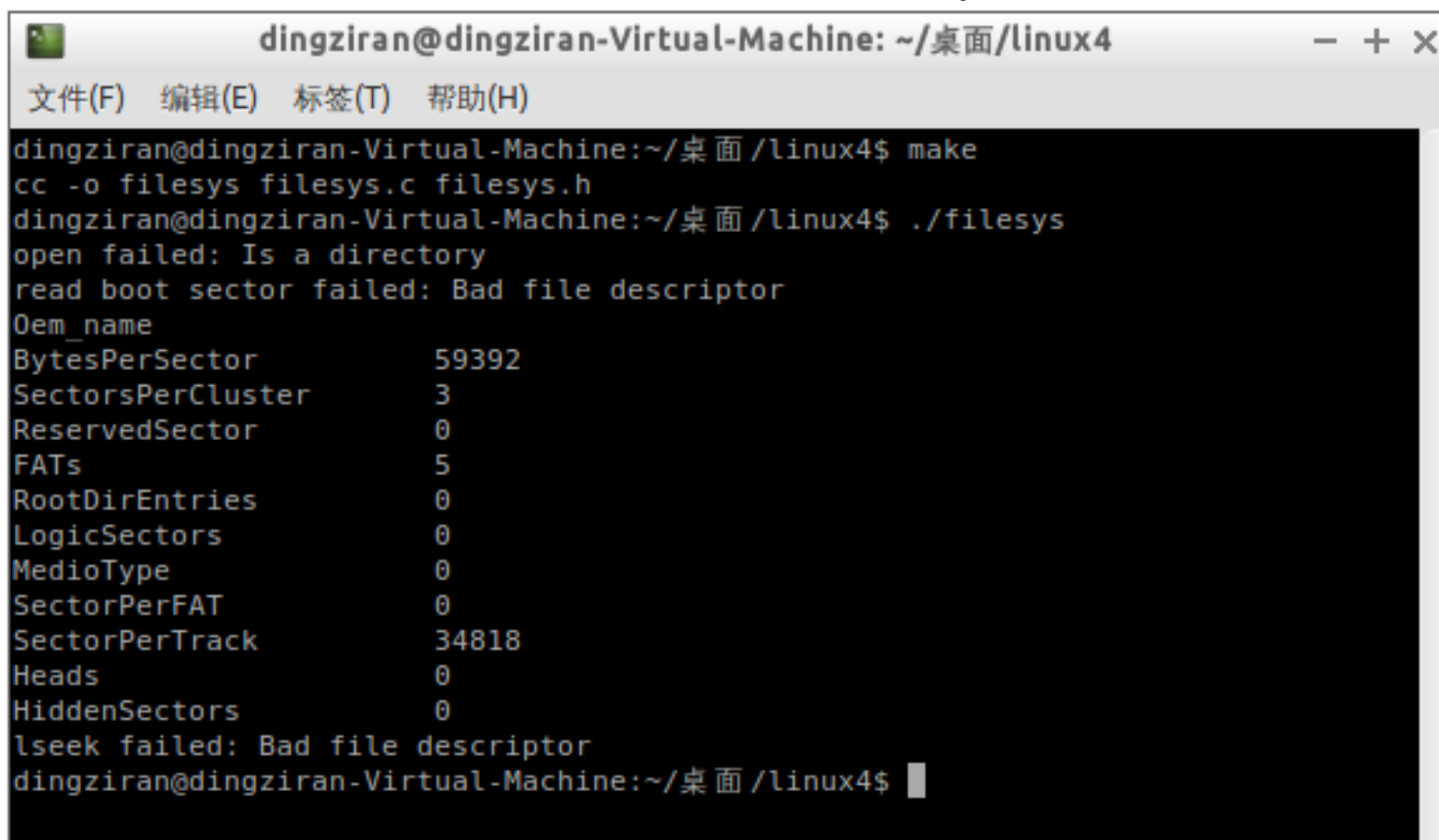
- 使用`sudo mkdir /dev/sdb1`创建这个目录
- 然后使用`sudo mount -o loop data /dev/sdb1`命令进行挂载
- 现在可以文件夹进入`/dev/sdb1`这个目录了，就好像进入了优盘一样





## 4.运行源代码

- 执行make命令，然后执行./fileysys，程序成功编译执行





A terminal window titled "dingziran@dingziran-Virtual-Machine: ~/桌面/linux4" with standard window controls. The terminal shows the following commands and output:

```
dingziran@dingziran-Virtual-Machine:~/桌面/linux4$ make
cc -o fileysys fileysys.c fileysys.h
dingziran@dingziran-Virtual-Machine:~/桌面/linux4$ ./fileysys
open failed: Is a directory
read boot sector failed: Bad file descriptor
Oem_name
BytesPerSector          59392
SectorsPerCluster       3
ReservedSector          0
FATs                    5
RootDirEntries           0
LogicSectors             0
MedioType                0
SectorPerFAT             0
SectorPerTrack          34818
Heads                    0
HiddenSectors            0
lseek failed: Bad file descriptor
dingziran@dingziran-Virtual-Machine:~/桌面/linux4$
```

Step1

# 进行第一次提交

- 当出现 ，表示这个地方是检查点，做到这里需要进行提交，当最后完成实验将代码上交之后，助教会根据这些检查点进行打分，所以如果没有提交检查点，就没有相应分数。
- 当遇到  类似的标记，需要在项目根目录下，即linux4目录，执行`git add .`（这是一个点），然后执行`git commit -m “学号+Step1”`
- 比如`git commit -m “11060000Step1”`
- 注：只有源文件和上次提交有差异，才能成功提交

- 如果提示unable to auto detect email, 则根据提示
- 执行以下两个命令

```
git config --global user.email "you@example.com"  
git config --global user.name "Your Name"
```

- 之后提交, 以后可以通过**git log**命令查看提交历史

```
nothing to commit, working directory clean  
dingziran@dingziran-Virtual-Machine:~/桌面/linux3$ git log  
commit 286752e9e4b15787f83805f8adb40a0958bb79f5  
Author: dingziran <dingziranrr@163.com>  
Date: Tue Apr 29 10:59:37 2014 +0800  
  
11060000Step1  
dingziran@dingziran-Virtual-Machine:~/桌面/linux3$
```

- 如果提示unable to auto detect email, 则根据提示
- 执行以下两个命令

```
git config --global user.email "you@example.com"  
git config --global user.name "Your Name"
```

- 之后提交, 以后可以通过git log命令查看提交历史

```
nothing to commit, working directory clean  
dingziran@dingziran-Virtual-Machine:~/桌面/linux3$ git log  
commit 286752e9e4b15787f83805f8adb40a0958bb79f5  
Author: dingziran <dingziranrr@163.com>  
Date: Tue Apr 29 10:59:37 2014 +0800  
  
11060000Step1  
dingziran@dingziran-Virtual-Machine:~/桌面/linux3$
```

- 这样做的目的是方便进行检查，可以从提交的日期、姓名内容等，大概了解到完成的情况，然后助教再还原到相应检查点就可以检查每一步，注意，检查点检查的不是截图，而是你的代码改到运行结果和截图相似时候的代码的状态。
- 之后ppt看到step的圆圈就不再进行说明，请参考前几页的说明
- 每次做完用git log命令查看一下，自己是否已经完成相应检查点
- 之后相应实验的截止时间为那次实验最后一节课当天发助教邮箱

# 修改bug

- 打开filesys.c，建议使用gedit工具，当然使用其他也可以，打开后找到main函数，发现第一次运行的错误是由于打开文件错误，open命令不能打开一个文件路径，所以打开filesys.h，修改DEVNAME为

```
#define FILESYS_H  
#include<stddef.h>  
#define DEVNAME "data"  
#define DIR_ENTRY_SIZE 32  
#define SECTOR_SIZE 512
```

- 使用make命令重新编译运行

```
cc -o filesystem filesystem.c filesystem.h
dingziran@dingziran-Virtual-Machine:~/桌面/linux4$ ./filesystem
Oem_name           mkdosfs
BytesPerSector      512
SectorsPerCluster   4
ReservedSector      1
FATs                2
RootDirEntries      512
LogicSectors        2048
MedioType           248
SectorPerFAT         2
SectorPerTrack       32
Heads                64
HiddenSectors        0
please input a command, including followings:
    ls                list all files
    cd <dir>           change direcotry
    cf <filename> <size> create a file
    df <file>          delete a file
    exit              exit this system
```

>



Step2



# 运行这些基本命令

```
HiddenSectors      0
please input a command, including followings:
    ls                list all files
    cd <dir>          change direcotry
    cf <filename> <size> create a file
    df <file>         delete a file
    exit             exit this system

>ls
Root_dir
      name      date            time            cluster size      attr
>cd .
>cf demo 1024
>ls
Root_dir
      name      date            time            cluster size      attr
      DEMO      2107:15:31      0:0:0          2            1024            file
>df demo
>ls
Root_dir
      name      date            time            cluster size      attr
>exit
dingziran@dingziran-Virtual-Machine:~/桌面/linux4$
```

# 本周内容和下周内容

- 发现基本命令基本正常，但是界面上就能看到各种bug，比如date都2107年了
- 本周内容为阅读源代码，了解程序运行机制。边了解边修改bug。
- 下周内容为讲解程序运行机制，布置接下来的检查点。