**操作系统课程设计实验报告**

——实验四：文件系统

负责人姓名：张然殊

学号：14061146

日期：2016.5.21

**小组成员**

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 姓名 | 学号 | 实验分工 |
| 1 | 韩慧敏 | 14061218 | 实验一 |
| 2 | 张慧昕 | 14061117 | 实验二 |
| 3 | 李雨霜 | 14061177 | 实验三 |
| 4 | 张然殊 | 14061146 | 实验四 |

目录

[1.实验目的 4](#_Toc446001831)

[2.需求说明 4](#_Toc446001832)

[2.1基本要求 4](#_Toc446001833)

[2.2 提高要求 4](#_Toc446001834)

[2.3 完成情况 4](#_Toc446001835)

[3.设计说明 5](#_Toc446001836)

[3.1 程序流程图 5](#_Toc446001837)

[3.2基本要求实现说明 5](#_Toc446001838)

[3.3 提高要求实现说明 5](#_Toc446001839)

[4.收获和感想 5](#_Toc446001840)

# 1.实验目的

1.了解文件管理系统的作用和工作方式。

2.了解FAT文件系统的结构

3.学习文件管理系统的一般开发方法。

1. **需求说明**

## 2.1基本要求

1.设计并实现一个目录列表函数（无须支持选项，如ls -a、ls -l等），用来显示当前目录下包含的文件信息。

2.设计并实现一个改变目录函数（无须处理路径名，如../../directoryName等），用来把当前目录切换到上一层目录或当前目录的子目录。

3.设计并实现一个删除文件函数，使用当前目录中的要删除的文件名作为参数，用来删除指定的文件，要注意文件的隐藏、只读和其他系统属性。

4.设计并实现一个创建文件函数，使用要创建的文件名和文件大小作为参数，用来创建一个新的文件。

## 2.2 提高要求

1.增加创建目录的功能。

2.增加删除目录的功能：通常需要先判断目录是否为空目录，若目录不为空，则需给出提示，并删除其包含的所有子目录和文件；若是空目录则可以直接删除。

3.增加绝对路径和多级目录的支持：这里需要对输入的目录路径字符串进行解析，然后逐级查找目录。

4.对ud\_cf()函数进行改进，使其可以向文件中写入实际内容，并根据写入的内容计算文件的实际大小。

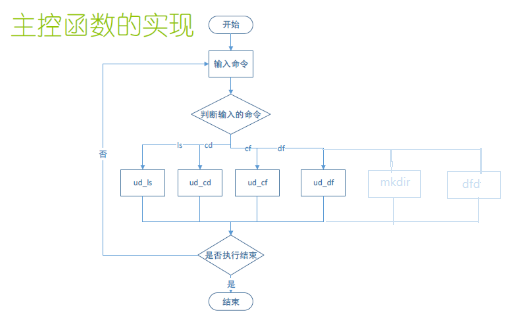
5.对ud\_ls()函数进行改进，增加对全部非根目录信息的读取（基本要求中仅读取一个扇区的非根目录细信息）。

## 2.3 完成情况

完成了以下功能：

1. 实现一个目录列表函数
2. 实现一个改变目录函数
3. 实现一个删除文件函数
4. 实现一个创建文件函数
5. 增加创建目录的功能
6. 增加删除目录的功能
7. 增加绝对路径和多级目录的支持
8. 对ud\_cf()函数进行改进
9. 对ud\_ls()函数进行改进
10. **设计说明**

**3.1 程序流程图**



**3.2 基本要求实现说明**

1.实现一个目录列表函数

2.实现一个改变目录函数

3.实现一个删除文件函数

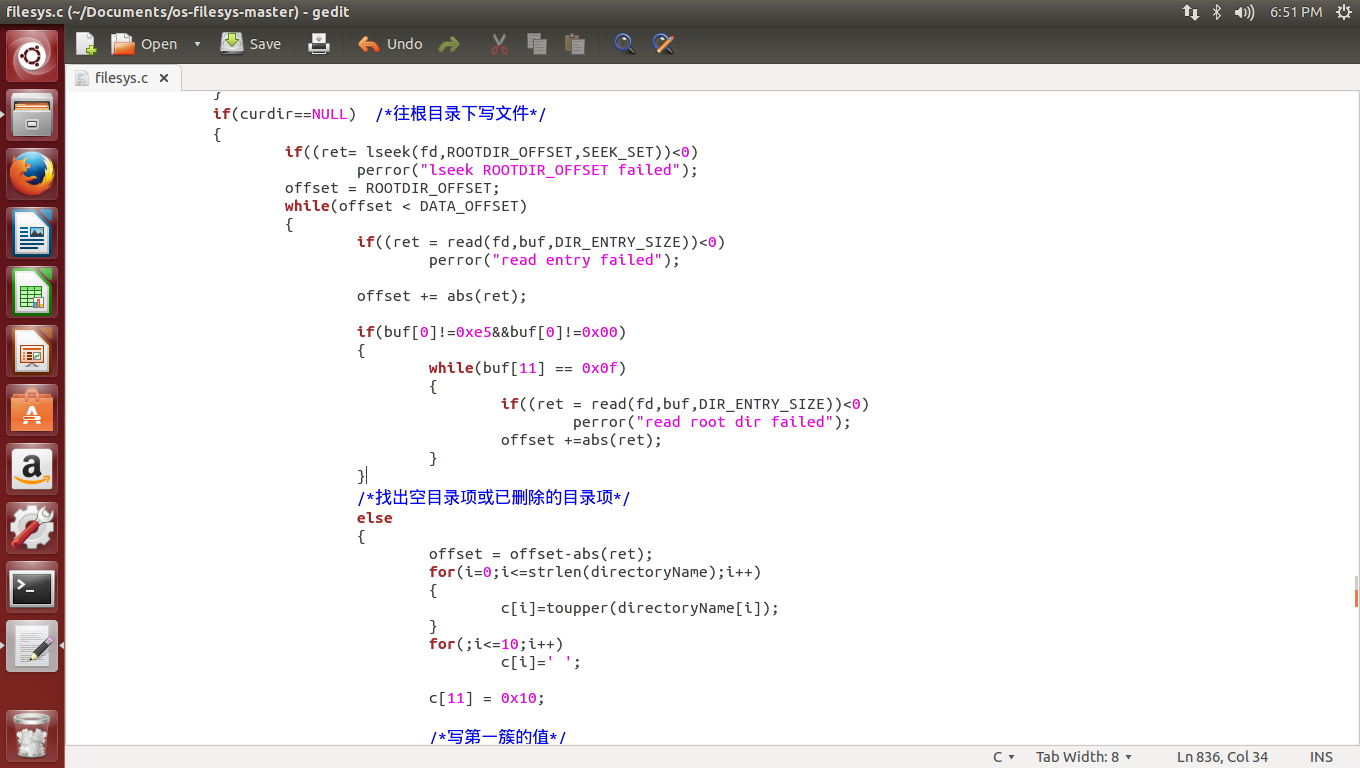
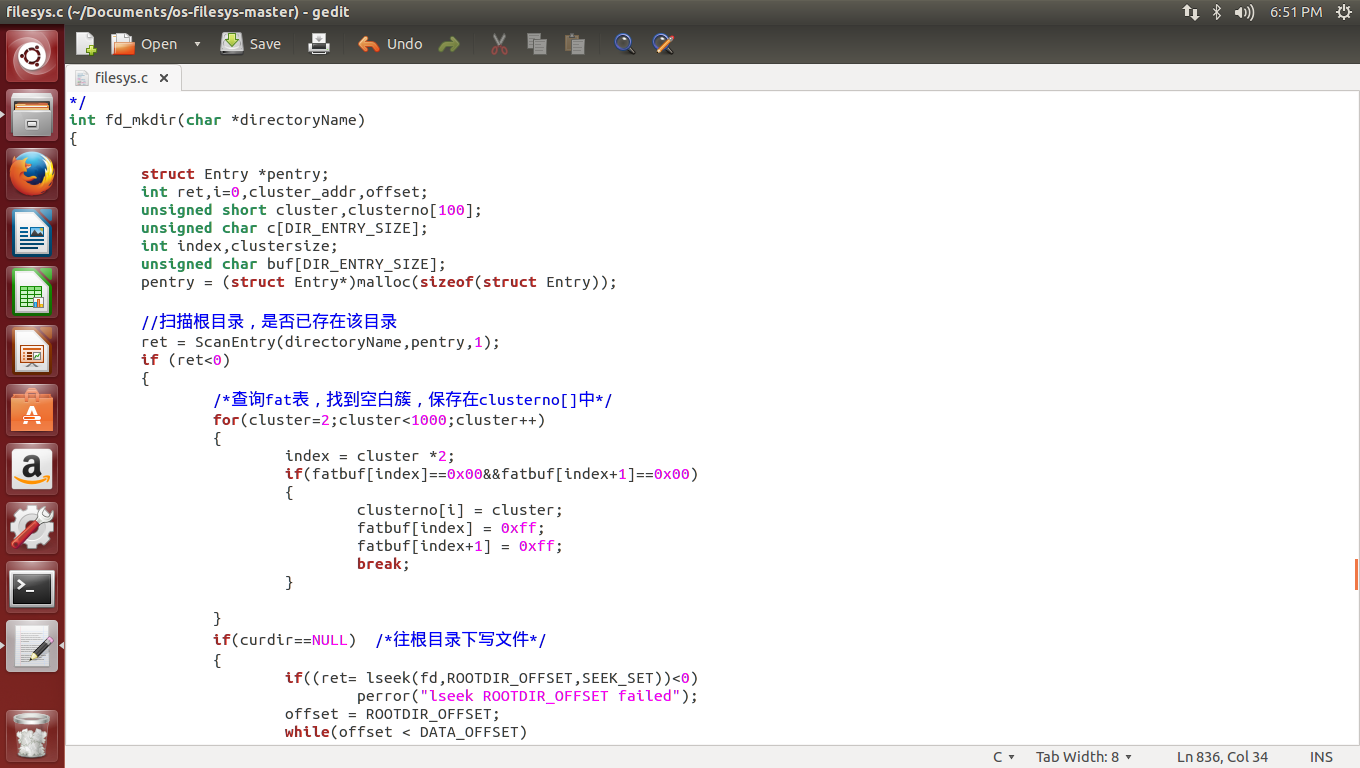
4.实现一个创建文件函数

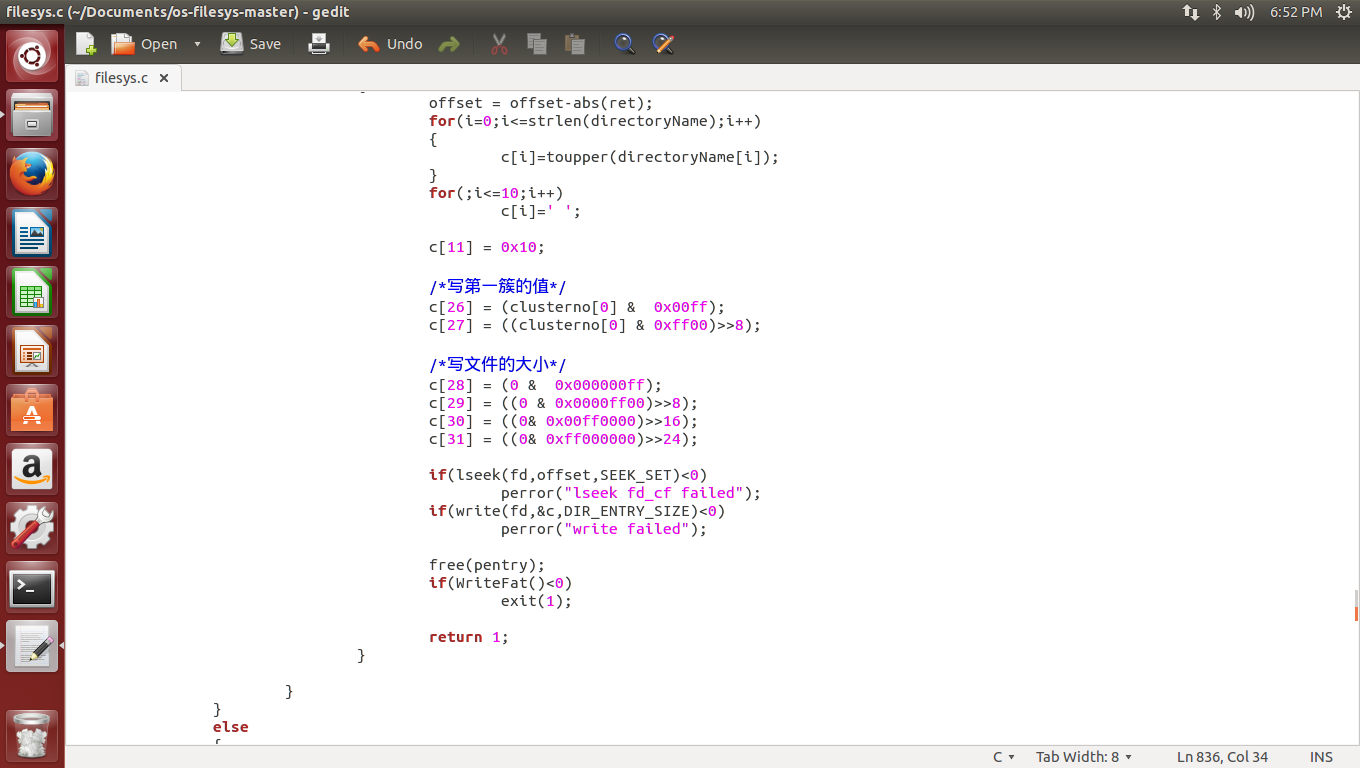
## 3.3 提高要求实现说明

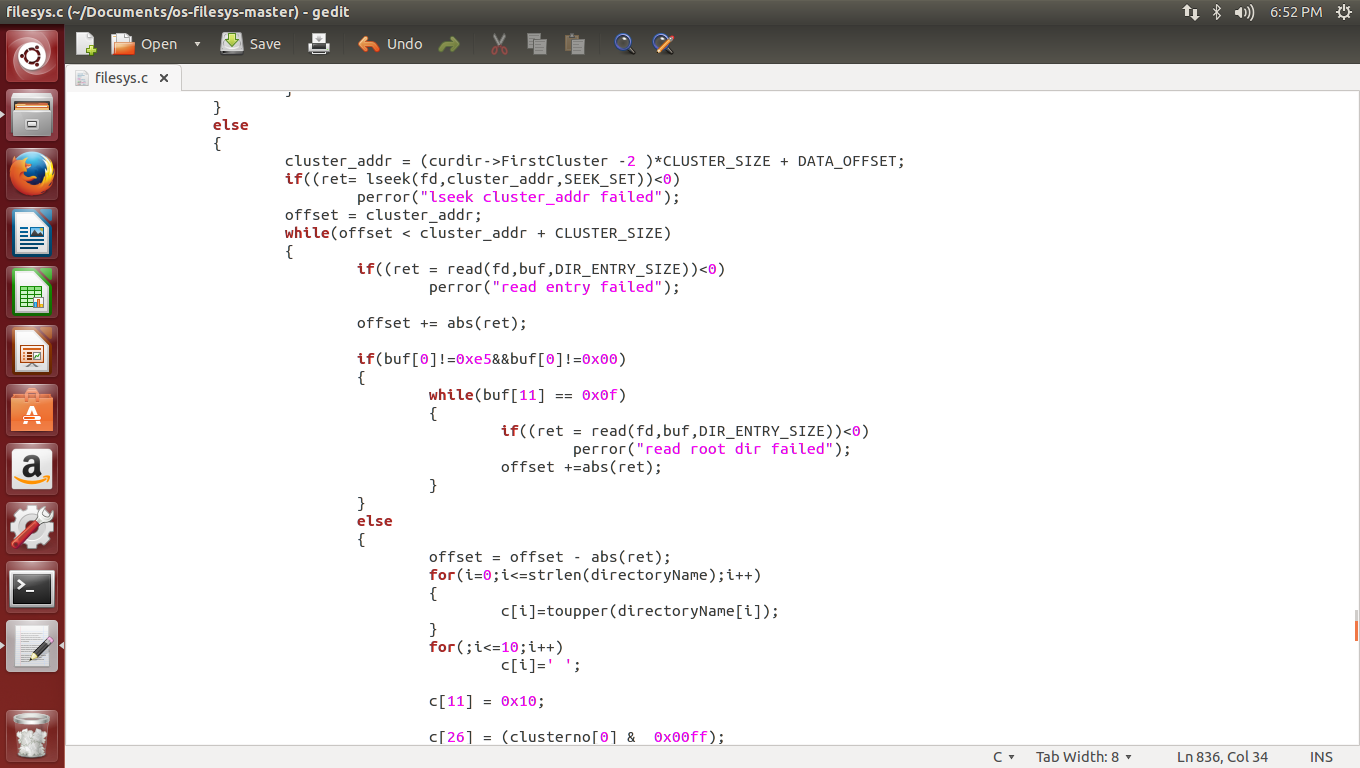
1. 创建目录

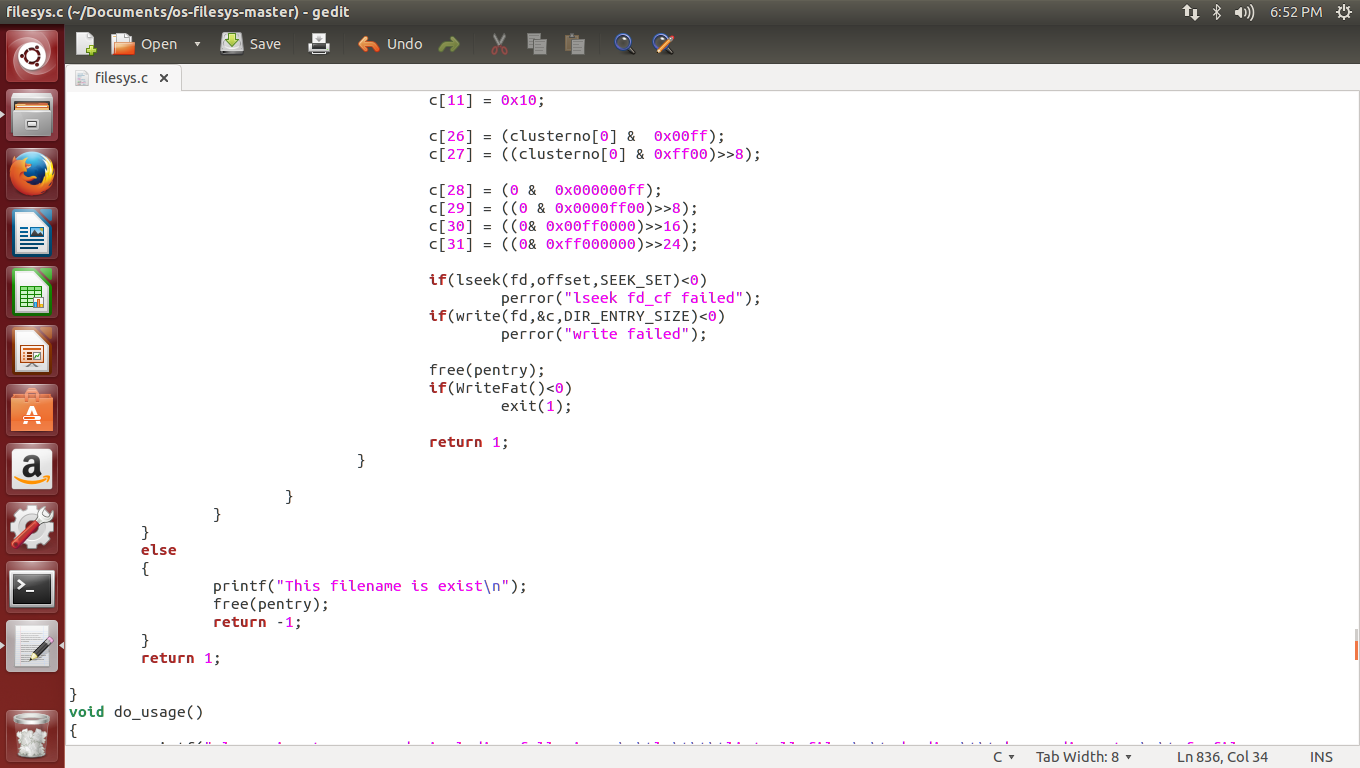
参考源代码中创建文件的内容。目录事实上和文件是同一性质的，只是表现出来的属性为目录，因此在源代码上的改动不是很多，主要是在调用函数ScanEntry时涉及参数传递等问题。在此基础上，我们是能够支持多级目录的。

此部分代码如下：



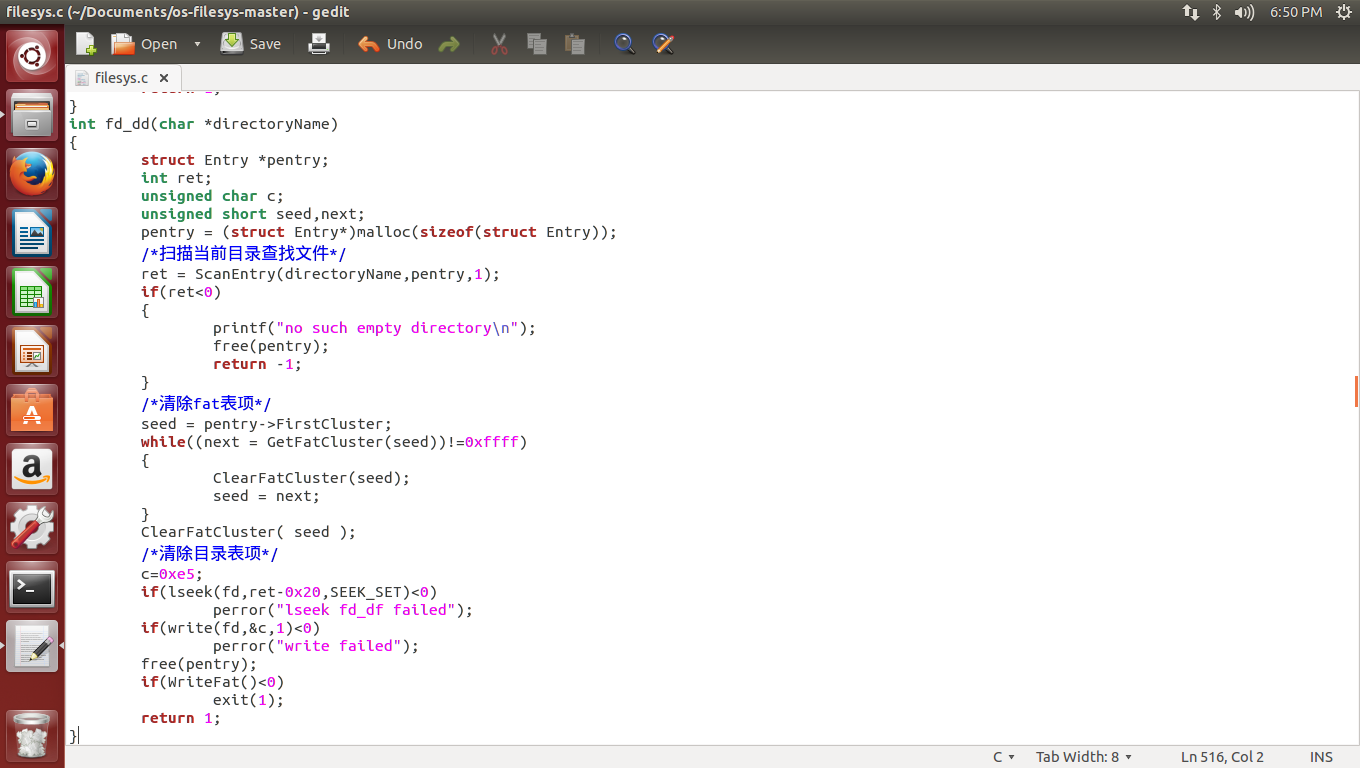




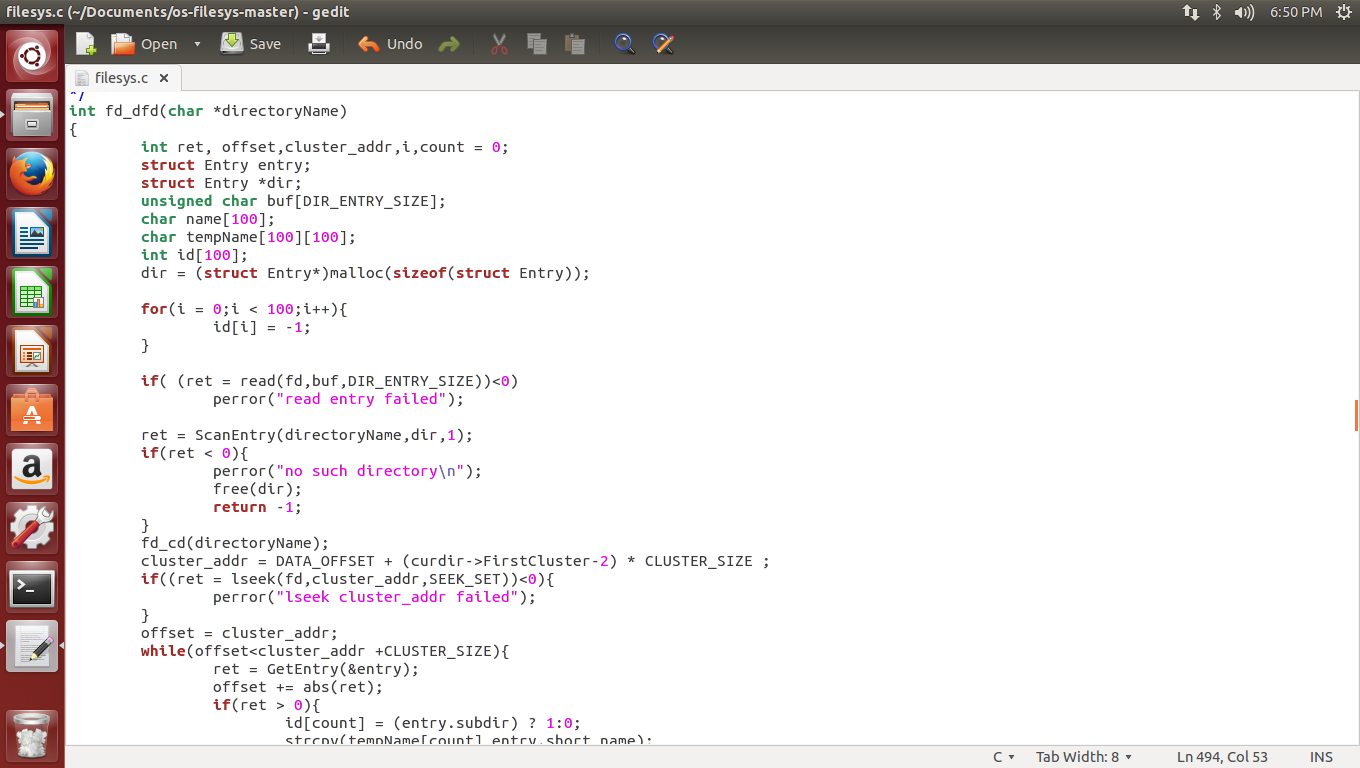


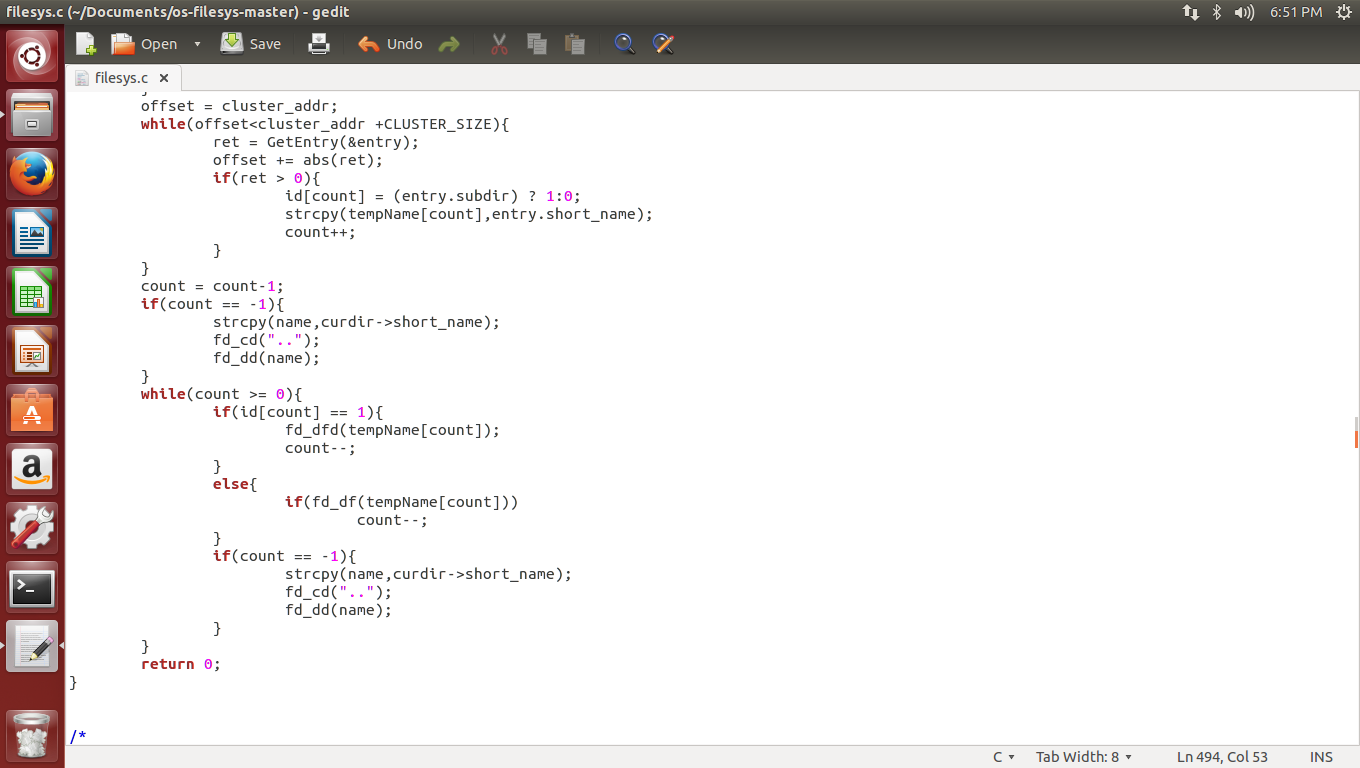
## 2.删除目录

参考源代码中删除文件部分的内容。新增了一个删除单个目录时的函数fd\_dd，由于目录事实上和文件是同一性质的，只是表现出来的属性为目录，此部分较源代码删除文件部分改动不大。



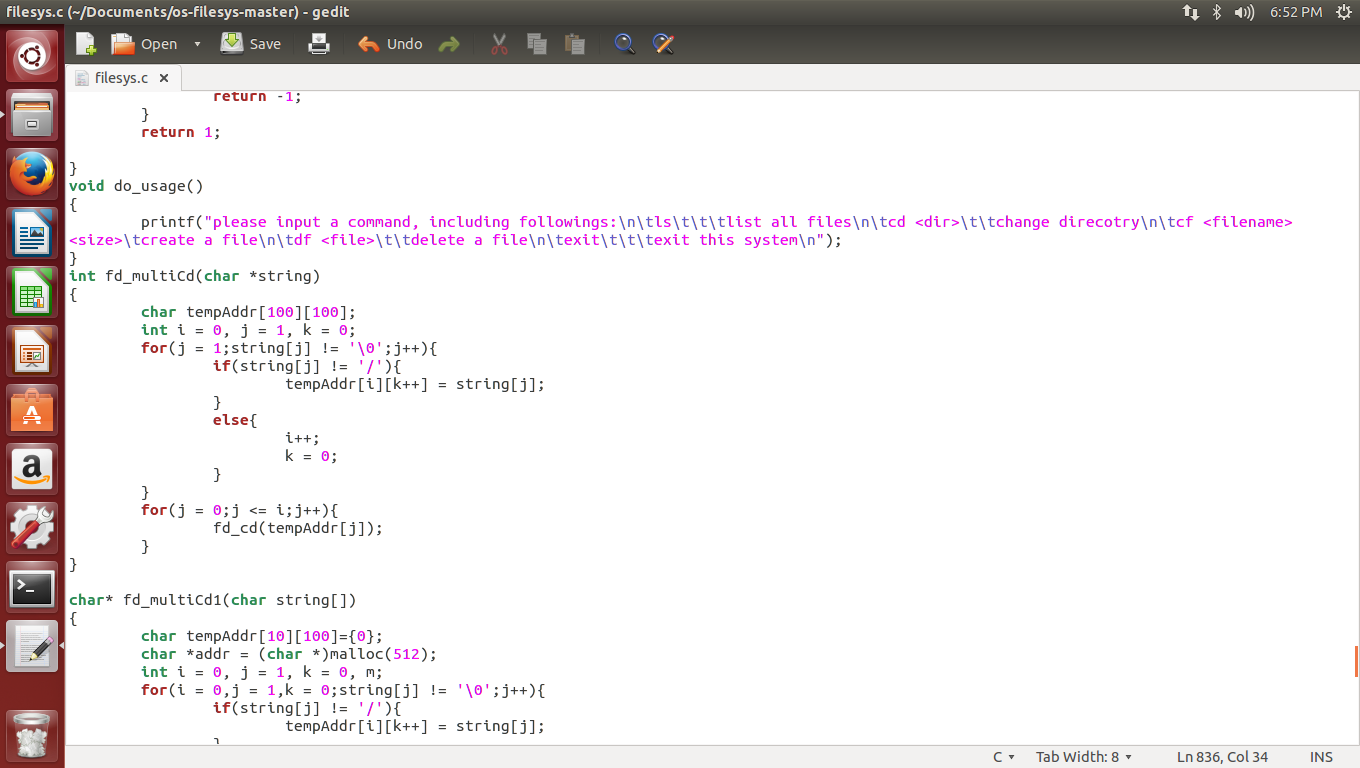
在fd\_dd基础上，我们设计了一个递归删除目录的函数fd\_dfd，用来删除目录下的所有内容：若为文件，则调用fd\_df删除，否则若为目录，则调用本身对于此目录继续递归删除，直至目录为空，再将之删除。代码如下：

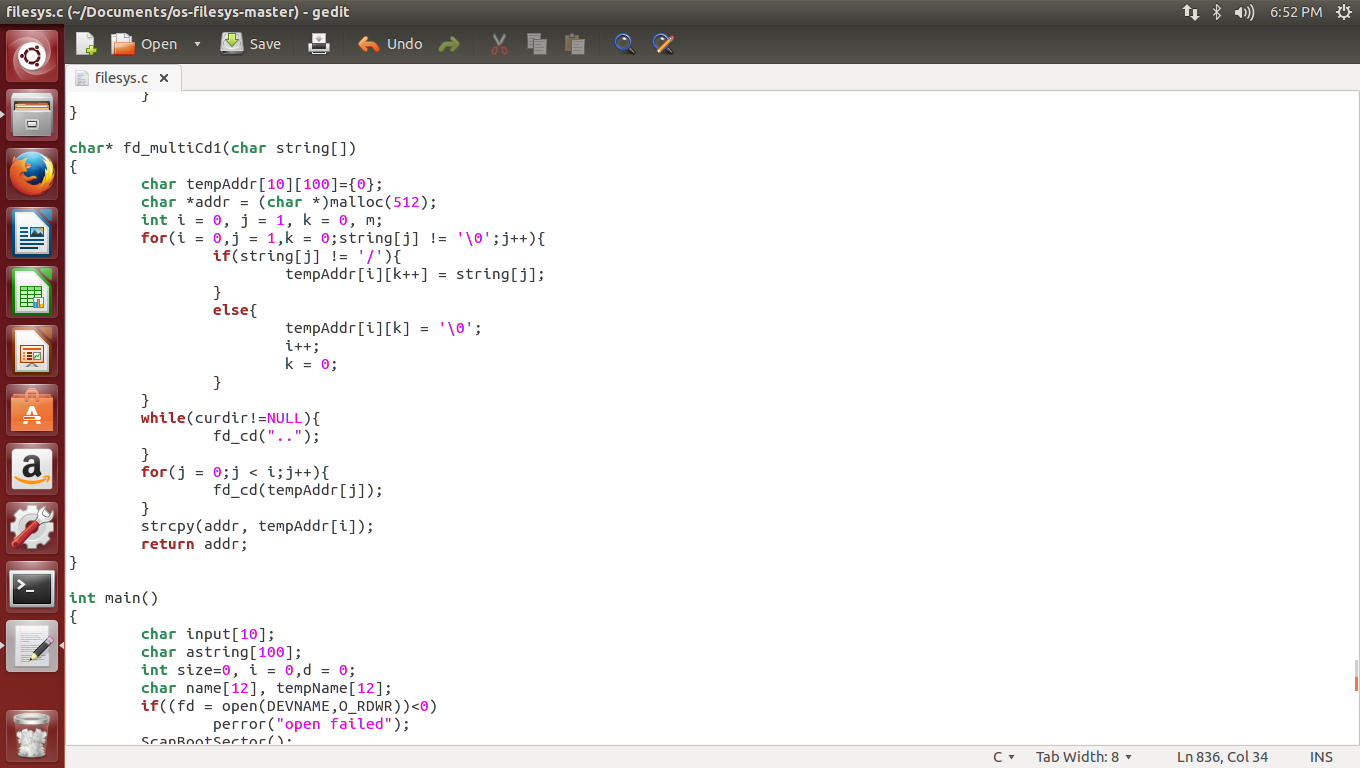




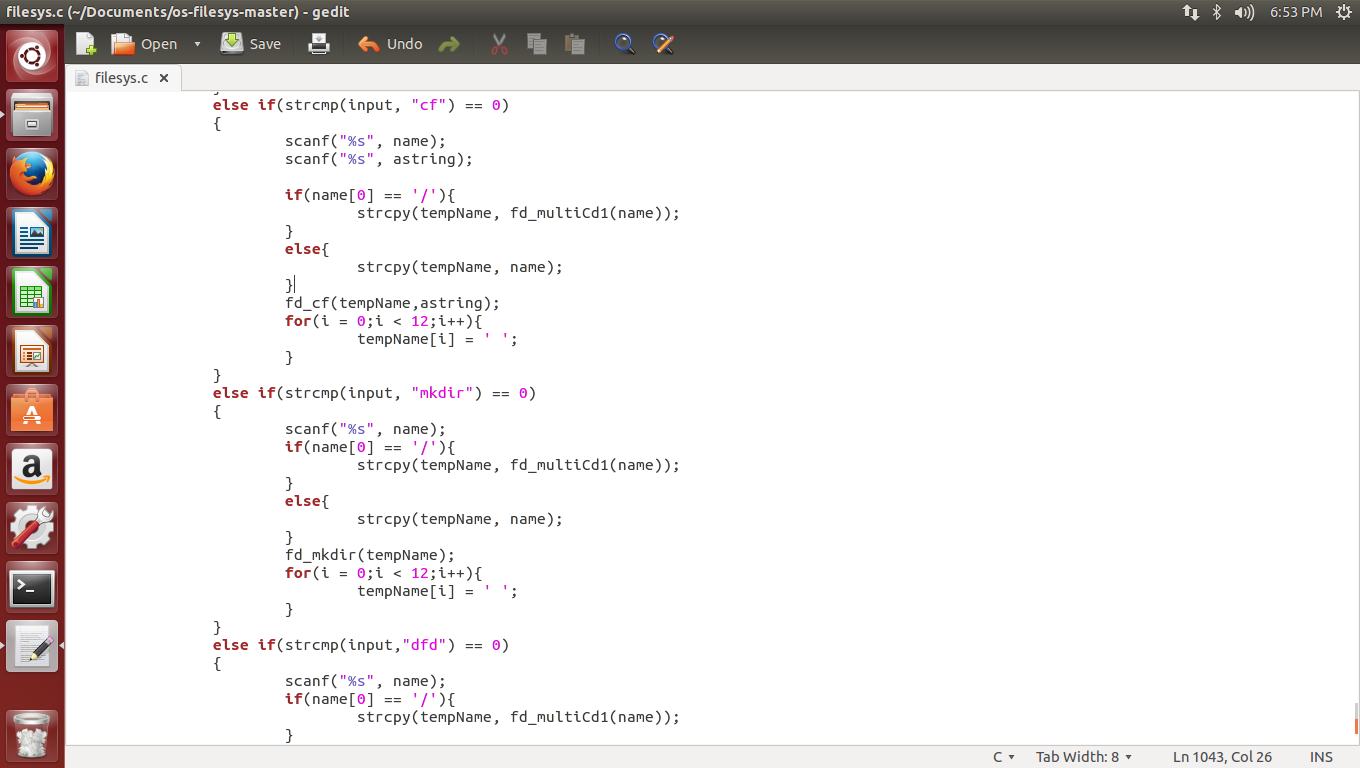
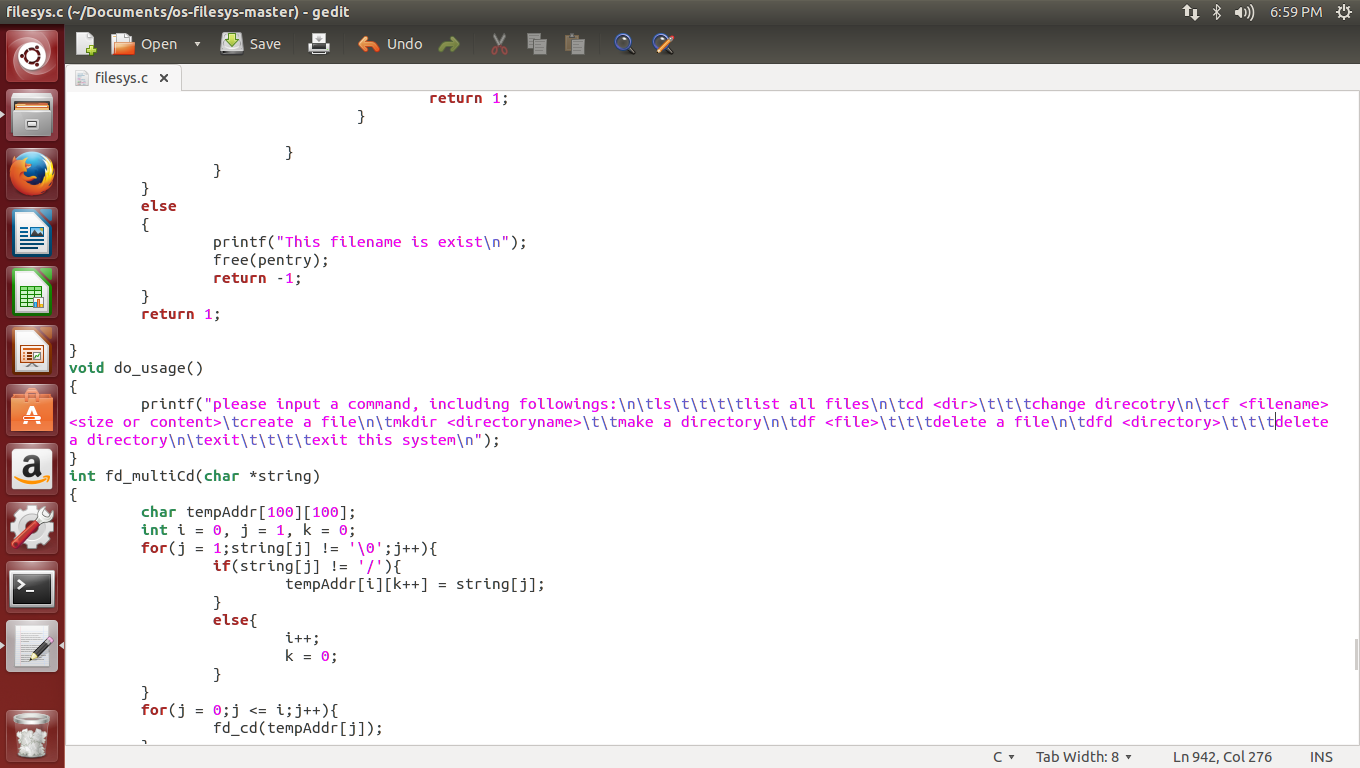
3.对绝对路径的支持

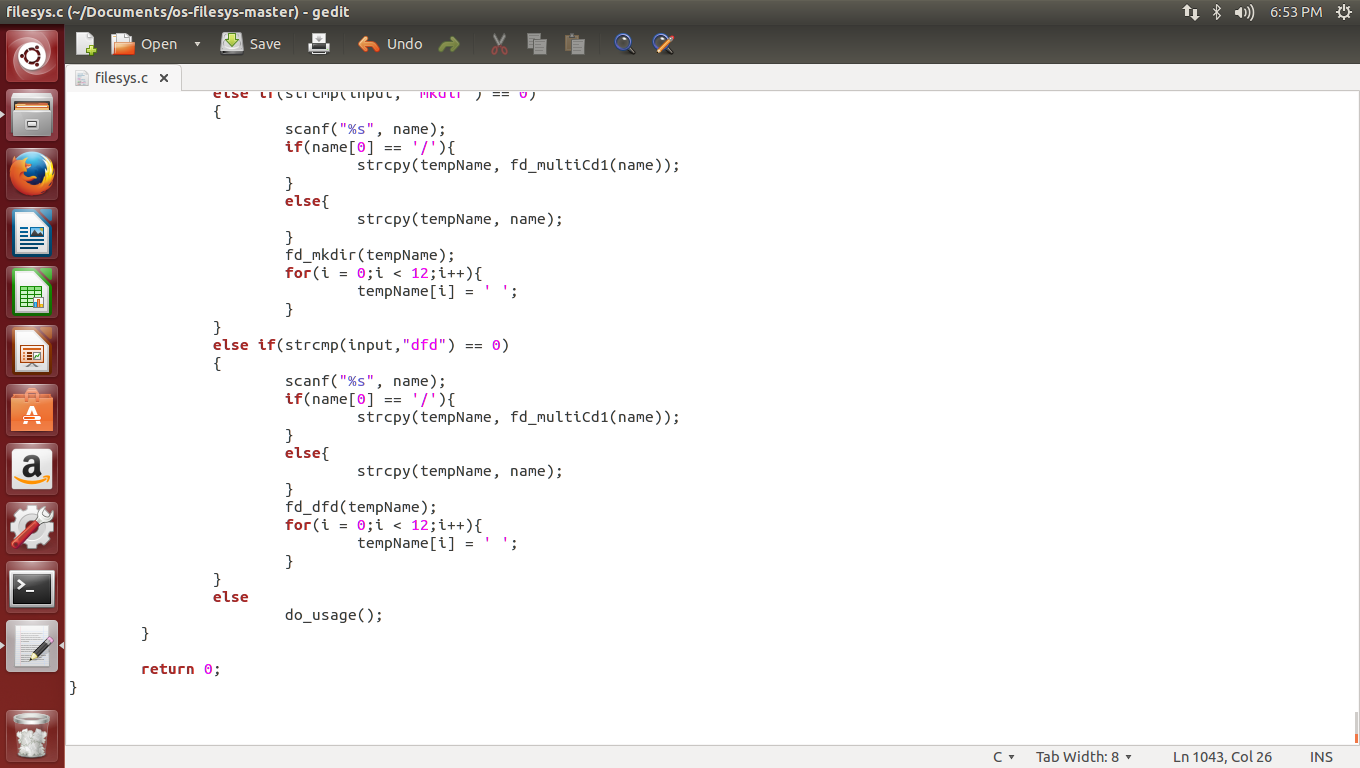
我们首先实现了两个函数 fd\_multiCd和fd\_multiCd1用于将读入的绝对路径转化为相对路径，在其中我们利用了fd\_cd函数实现路径的不断转移。在cd指令和其他指令使用了绝对路径的时候，我们将分别调用fd\_multiCd和fd\_multiCd1这两个函数用来实现路径的转移。代码如下：





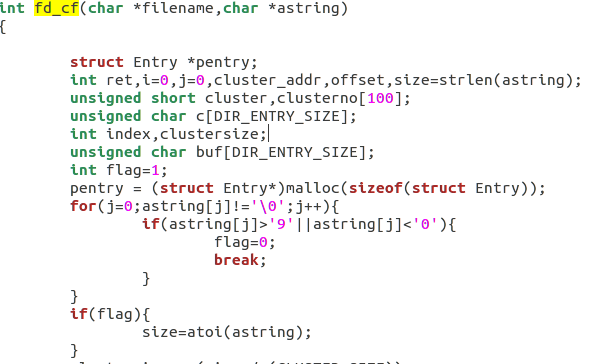
4.由于以上的改动，在do\_usage以及main中都做了相应改动，代码如下：





1. 对fd\_cf()函数进行改进

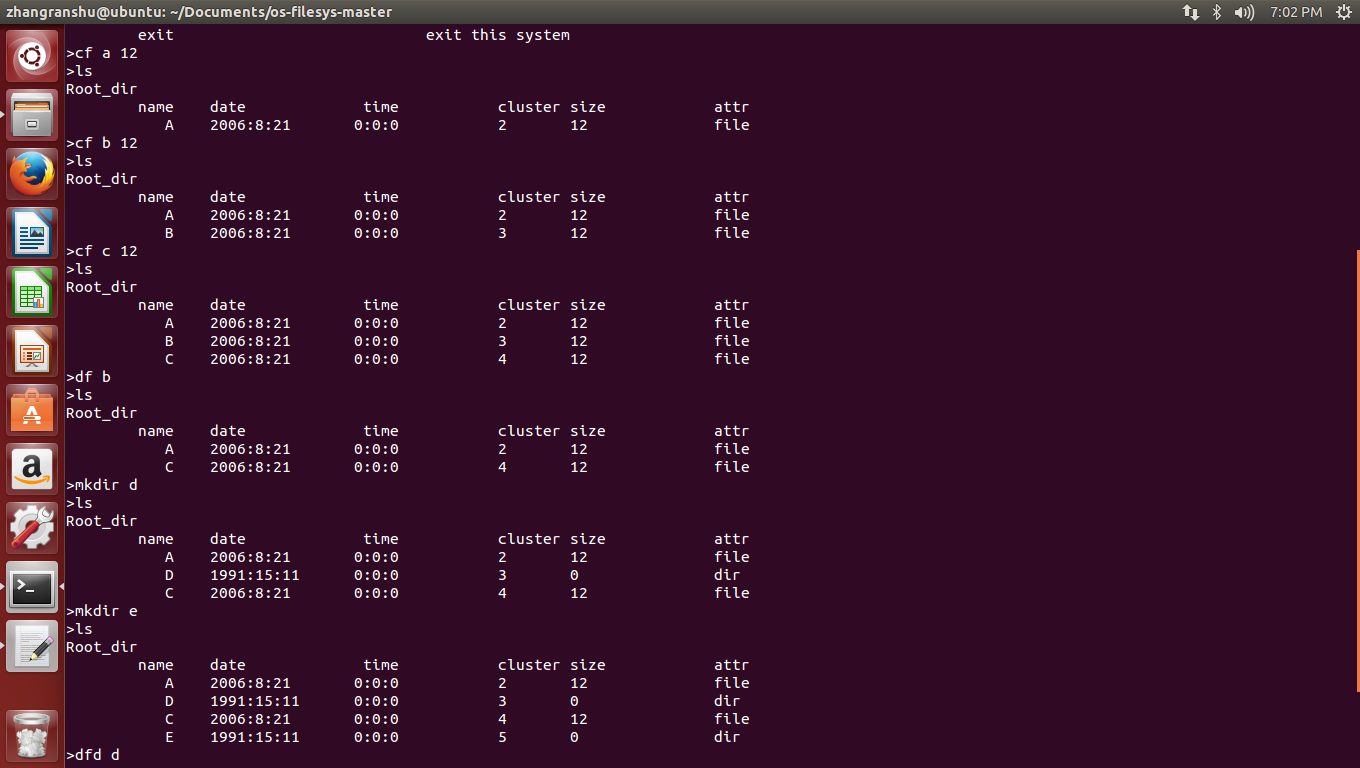
改变fd\_cd的第二个参数为字符串，size大小即为字符串长度（char每个占1字节），且进行判断若字符串的每一个字符均为数字，则认为输入的size大小，否则size为字符串所占大小。代码如下：

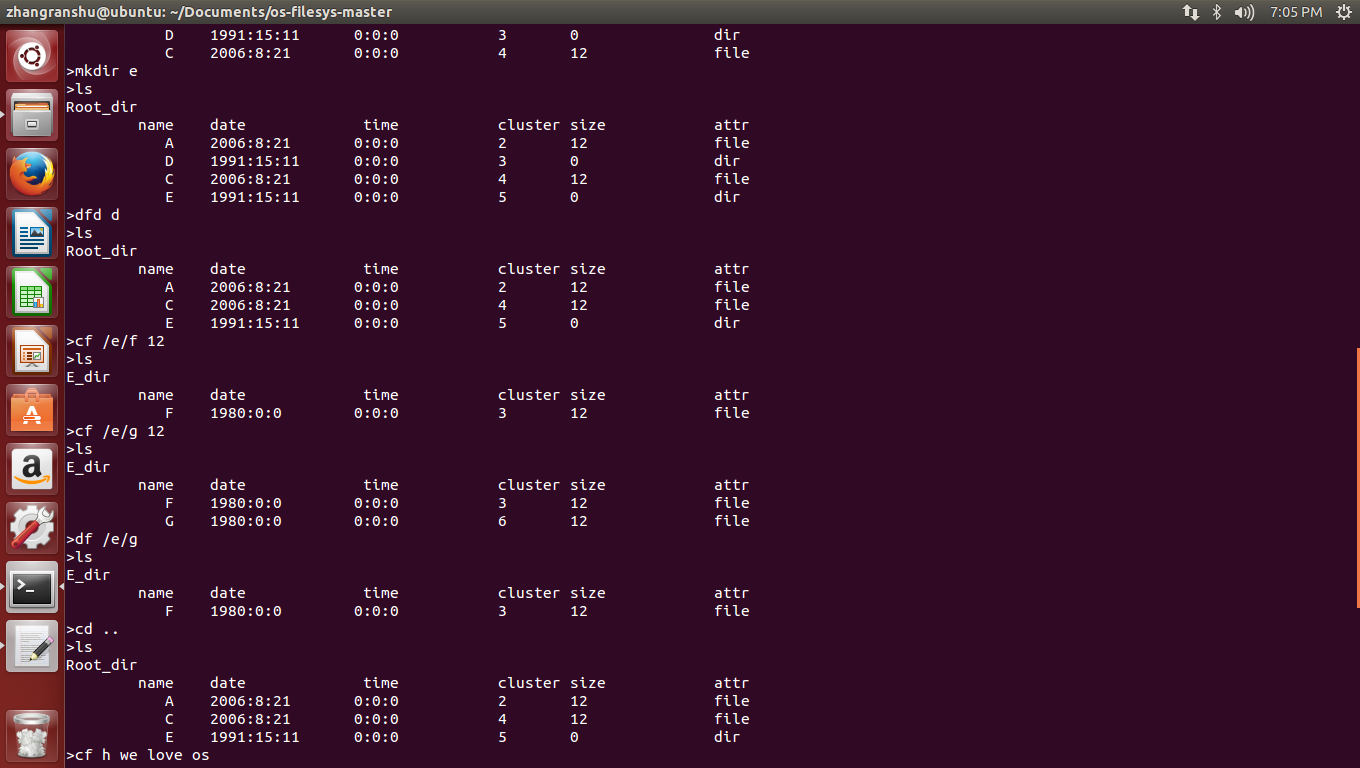


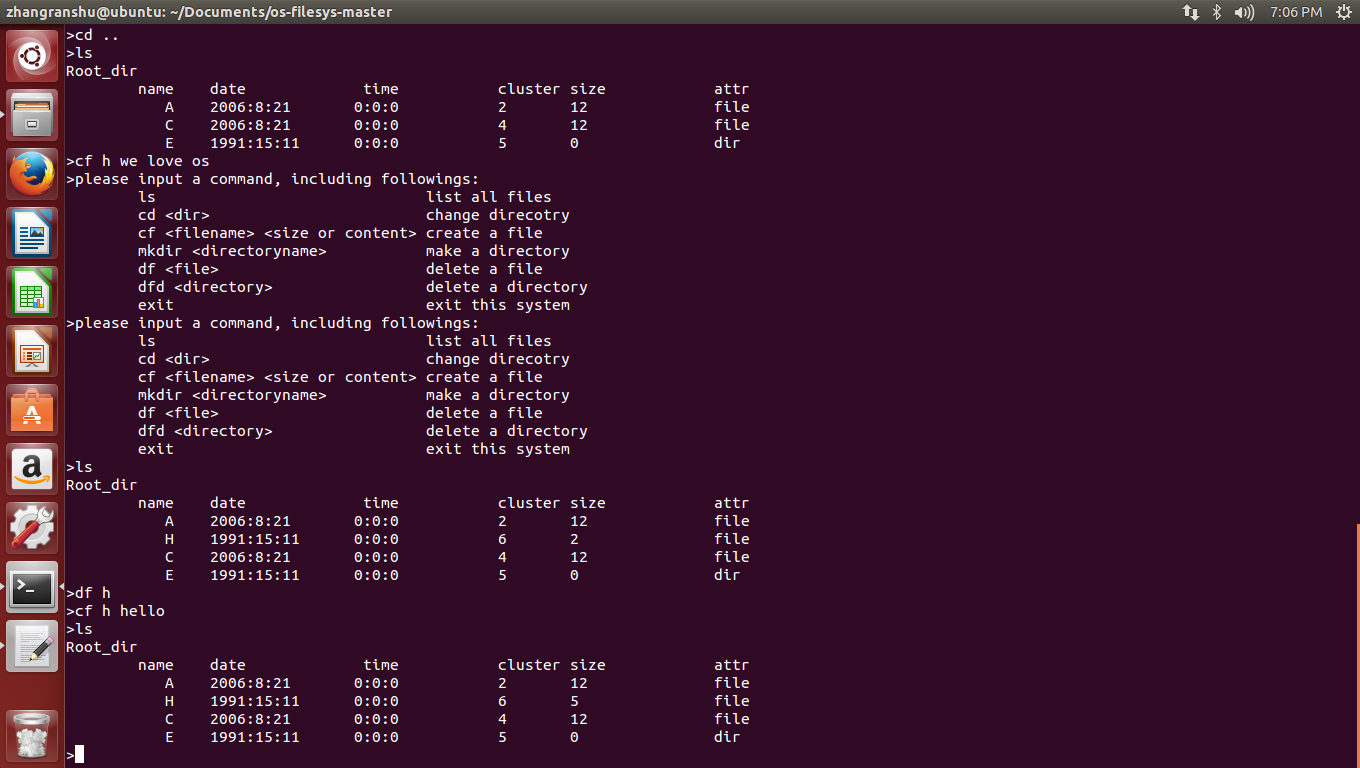
6.对fd\_ls()函数进行改进

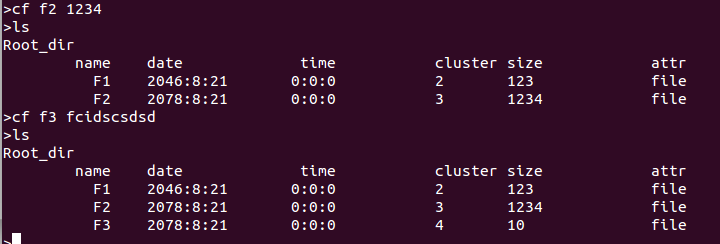
首先要将ls的非根目录访问从一个扇区（512）改进。我们认为访问一个簇（512\*64）足够了，如此可以访问1024个目录项。将代码中的SECTOR\_SIZE改为CLUSTER\_SIZE。

6.一些运行样例









# 4.收获和感想

李雨霜：本次实验的基础知识较为繁杂，如果对文件系统的基础知识了解不足很容易对源码就产生误解，也无法理解bug的行为。所以本次实验主要花费时间在研习书本、了解知识上。对ls和cf的修改，逻辑上都较为简单，但是容易发生偶发性的细节错误，主要的debug时间就花在了这上面。

韩慧敏：本次os实验其实并不算很难，关键在于理解FAT文件系统的工作原理，怪自己上课时候没有认真听讲，最开始做的时候完全是一头雾水，代码根本看不懂，就连PPT也是一知半解，看实验指导书，也还是不太明白具体的FAT表项，文件分配表等等的概念，总是混为一谈，去网上搜索，得到的结果也是鱼龙混杂，走投无路之时，看到实验指导书上的第四章有一部分关于FAT文件系统的详细讲解，总会总算是弄明白了FAT文件系统的工作方式和运行的原理，做实验的过程中还发现自己这段时间一直使用java语言，对C语言的使用十分手生，连二维字符数组的初始化都不会了，各处搜索，费了好大的力气，走了不少的弯路，看来学习新知识的时候也要记得复习旧知识啊！

张然殊：这次实验主要是针对文件系统进行考察，不得不说由于课上听讲效率有些偏低导致了在具体实验时遇到了一些问题，好在大家都很认真，一直在努力。本次实验的代码部分可以说部分是参考了源代码的，这让我们在生了一些力气的同时，也认识到自己代码编写能力的欠缺。说到代码编写能力，这次可以说是深有体会，由于C语言中指针运用不熟练，在编写函数时出现了许多问题。整体来说，这次实验在深化了课堂知识的同时也提高了我们的代码编写能力，尤其是磨练了我们的意志。感谢一起努力的同伴们！

张慧昕：这次实验需要文件系统的知识，然而对于老师上课讲过的知识，听得不是很认真，课下没能及时的复习，掌握的不牢固，不过多亏了小组制度，给了大家一个合作的机会，互相帮助，互相扶持，能够更好的完成作业。