OS第四次实验报告

1. 实验内容：FAT16文件系统的管理，功能包括
   1. 列出目录项（ls），支持显示修改时间、文件大小、起始簇号、文件类型（文件/文件夹）等信息，支持大小超过一簇的长目录；
   2. 切换当前目录（cd dir1/dir2/dir3），支持绝对路径和多级路径；
   3. 创建文件并写入内容（cf name content）；
   4. 创建目录（mkdir name）；
   5. 删除文件或目录（df name），如果是目录，将先递归删除所有内容，再删除目录。
2. 实验分工，排名不分先后：
   1. 黄秋宇（14061176）：增加将文件内容写入簇功能
   2. 胡亚龙（14061196）：增加新建文件夹功能
   3. 李搏（14061181）：增加多级目录和绝对路径支持
   4. 林子义（14061194）：增加递归删除和长目录功能
3. 实验原理：
   1. FAT文件系统可以提供多级目录结构，如果一个文件是目录，则其中存储有其所有子文件的基本信息（每一条信息称为一个目录项），如果一个文件是普通文件，则其中存有内容。
   2. FAT16文件系统为每个簇配备一个指针，这个指针指向该簇的下一内容的所在簇号，这样，簇之间以链表的形式联系起来，使得文件可以在不超过总剩余容量的情况下任意长。指针集中存储在FAT表中，在磁盘上占用一块连续的区域，以供快速索引。第i簇的指针占用FAT表的第i项，i从2开始编号，因而FAT表的第0和1项不具有指针意义。FAT[i]=0x0000表示该簇未使用，FAT[i]=0xffff表示该簇没有后续簇。
   3. FAT的目录项定义参见相关文档。
4. 实现方法：
   1. 对于ls命令，只需要改变其遍历目录的方式，多加一层遍历来扫描所有的簇即可。需要对ScanEntry函数也做类似的修改，以便df和cf等功能能够在所有簇查找指定名称的文件是否存在。
   2. 对于cd命令，通过加入递归来支持多级路径，即每次调用先提取第一级目录，进入该目录，然后再对剩下的所有级递归调用cd命令。cd加多级路径以后有一个坑点，就是如果中间有某一级目录不存在，这样就需要恢复之前已经成功进入的目录，方法就是每次进入一级以后不立即退出，而等待后续递归的结果，如果成功则退出，如果失败则恢复本级的更改，并告知上级进入失败。这样每一级都将恢复当级的更改，从而能恢复到初始的情况。绝对路径的处理方法类似。
   3. 对于cf命令，首先更改了命令读入格式，读入的第二参数由长度变成内容。然后cf命令添加的主要功能就是分配簇以后，向其中写入内容。该功能的实现较为简单，就是标准的链表操作，但也有一个坑点，就是每次开一个文件，都要清空未填满位置的所有数据，因为在文件删除（甚至是快速格式化的时候）都不会清除簇中的数据，这样轻则导致文件末尾混入不明数据，更严重的问题是，如果剩余内容中有目录项，而所在簇恰好又是一个文件夹的一部分，这样对这个实际并不存在的目录项执行删除、进入等操作时就会发生混乱，会造成其他功能也出错。对于cf，其实还考虑写入以后当前文件夹大小超过当前分配簇的情况，这样还需要先扩大当前目录，再向新簇中写入内容。
   4. 对于mkdir命令，其实就是创建一个文件，属性设置为目录，然后向其中填入两个目录项（.和..），新填目录项的起始簇号为当前目录和上级目录（如果上级为root设为0），类型设置为目录即可。也就是说，相当于cf以后，向其中填入特殊的64bytes数据，所以实现原理不是很困难，但相关内容的填写方法需要阅读FAT16的说明文档。
   5. 对于df，主要是在删除主目录项的同时，要一并删除前面附带的长文件名目录项。注意到GetEntry函数自带有扫描前面有多少长文件名目录项的功能，因此只要将其返回值传递出来，并在主目录项之前多删除这么多项即可。这种做法可能处理不了一些特殊情况？
5. 运行效果在现场测试中已经有体现。
6. Bug修复：

现场的文件夹大小不为0、.和..属性为file问题均已解决