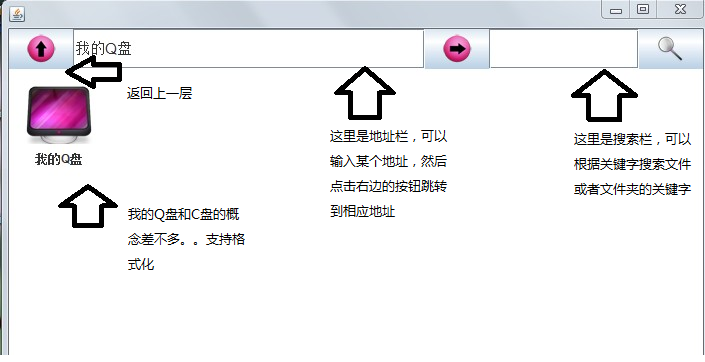
**文件系统设计报告**

1152745 邱峰

1. **程序使用说明**

打开程序的初始界面。



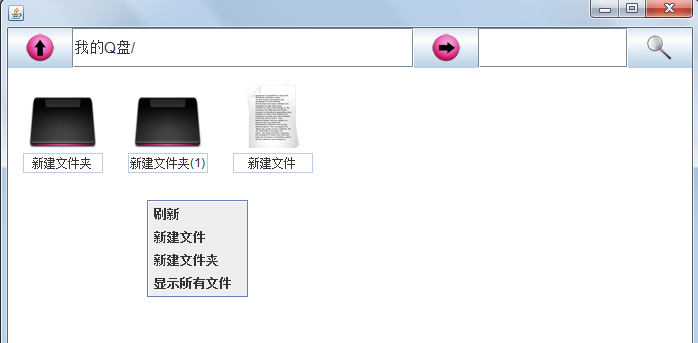
关于地址栏：

如果是 我的Q盘/新建文件 会自动跳转到相应的文件夹下打开文件

如果是 我的Q盘/新建文件/ 是打开文件夹

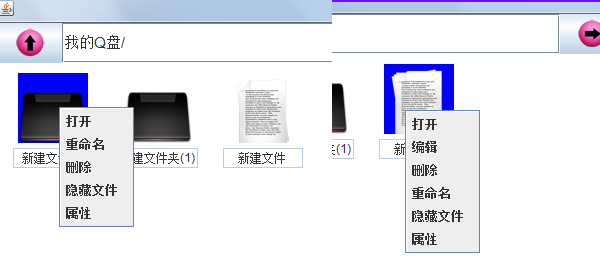
双击我的Q盘可以进入到里面（下面的文件与文件夹双击均可打开）

右键空白地方会显示出相应的属性，图为建立了2个文件夹和一个文件的界面

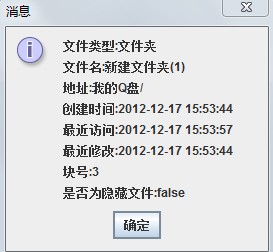


右击文件可显示与文件相关的操作

下图为文件和文件夹的不同操作

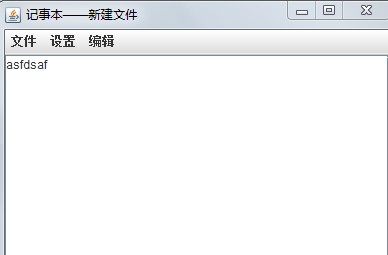


点击属性显示出一个文件夹或者文件的属性



打开文件夹进入下一层目录，和我的Q盘下的目录相同

代开文件，是自己编写的一个简易的记事本



在设置菜单下可以更改字体大小，编辑菜单中有包括撤消、重做、和替换的功能。

如果一个记事本已经被打开，然后又在外面点击删除时，会弹出警告框



同样，删除该文件所在的文件夹也会有相应警告。

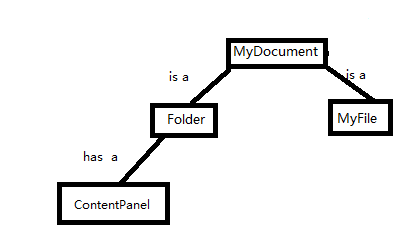
如果一个记事本被打开，然后又在外面打开时，不会出现2个记事本，正在打开的记事本会重新获取用户焦点，也就是跳到第一个窗口。

如果修改了文件，关闭时会提示你是否保存，当然也可以通过菜单栏下面的文件设置直接保存。

1. **程序设计图**

Ps：这个程序在设计上有一点失误，因此，设计图不是很好。

程序共有Disk（我的Q盘）、Folder（文件夹）、MyFile（记事本）三个主要的类。



MyDocument(FCB)抽象类:

该类中提供一些函数的接口，以及文件属性

主要的成员变量有

class MyDocument

{

ContentPanel contentPanel; //文件夹下的内容(file不会被实例化)

ContentPanel fatherContentPanel; //所在的文件夹

String whoAmI; //文件or文件夹

String fatherAddress; //所在文件夹的地址

String name; //名字（fatherAddress+name=绝对地址）

String createTime; //创建时间

String visitTime; //访问时间

String modifiTime; //修改时间

Block block; //用的块（只是一个指针，不会在这里new）

**boolean** isHide=**false**; //是否为隐藏文件

}

Folder类（文件夹）

继承自MyDocument

//contentPanel是运行哪个文件下的面板，比如在新建文件夹目录下，则界面显示的是新建文件夹.contentPanel

class ContentPanel

{

Vector<Folder> folderList; //该文件夹下的文件夹 以这个构成目录

Vector<MyFile> fileList; //该文件夹下的文件

ContentPanel fatherContentPanel; //所在的文件夹

**static** **boolean** *isShowAll*=**false**; //是否显示所有文件

**static** ContentPanel *runningPanel*=**null**;//正在显示哪个文件夹

}

//FolderPanel是在上一层中显示的面板，比如需要进入到新建文件夹中，就需要双击这个面板，这个面板是被加入到fatherContentPanel中

Class FolderPanel extends JPanel

{

……

}

//该类在内部处理后，以“打包”的形式在fatherContentPanel出现，返回的类的样子就是这样的，其中该panel提供一些双击 以及右击查看属性的接口。

MyFile类（文件）

继承自MyDocument

{

JFrame frame //记事本

FilePanel fileView; //和FolderPanel一样，打包出来是

}

Disk类（磁盘）

不得不说。。这个类是设计最失败的地方。

因为当初想的是我的Q盘只能有一个，所以该类所有的东西都是static

{

**static** JFrame *mainFrame*; //主要Frame

**static** JPanel *mainPanel*; //主要的Panel

**static** DiskPanel *diskPanel*; //

**static** ContentPanel *contentPanel*; //我的Q盘下的那个面板

**static** Fat *fat*=**new** Fat(); //fat

**static** Block []*block*=**new** Block[1000]; //所有的块都放入到我的Q盘中

}

设计失败之处就是因为把disk独立出来了。。。导致程序有许多地方都在特殊判断fatherContentPanel是不是Disk.ContentPanel。因此，加入了太多的补丁和重复代码。

Block类（块）

{

String property; //文件（夹）的一些属性

String data; //文件（夹）数据

**int** index; //在Disk中是第几个block  
}

这个设计失败的第二个地方。。。。起初是想把block的概念分成2块，一块是固定给文件使用，用户无法访问到，用户在文件夹或者文件中修改内容只是修改data。但是，后来发现用String类型就无法模拟每个块大小的问题了。。。关于这个问题。。我看到一个学长用char数组。。我感觉太那什么了。。。就最后还是一个文件一个块了。

Fat类

{

**final** **static** **int** *totBlock*=10;//由于这里采用位运算，所以实际的Block总数totBlock\*10

**int**[] useBlock=**new** **int**[*totBlock*]; **//hash判断每个块是否被使用，采用位运算**  
}

FolerTooBar

工具栏。。提供搜索、后对、根据绝对地址打开文件或者文件夹的功能。

1. **实现思想**
2. **创建文件（夹）**

首先向Disk申请block，如果block申请成功，则分配相应的block给文件夹，并进行初始化处理。如果block申请失败，告知用户内存被用光了。。。

1. **删除文件和文件夹**

先说删除文件吧： 首先要判断文件是否正在被使用，如果正在被使用，需要警告用户，删除失败。否则返回删除成功信息，并从block中移除该文件。回收利用block。

然后是文件夹：文件夹的删除比较麻烦，需要一层一层利用递归删除下面文件夹的各个文件，扫面一遍该folder下的vector<fileList>，利用删除文件的方法，将每一个文件删除，只有在所有文件都删除成功的情况下（删除文件中，如果文件正在被使用，删除失败），返回删除文件成功的信息。 在删除完文件以后，扫面所有的vector<folderList>再次删除文件夹，利用递归的方法，只有在所有的文件夹都被删除成功的情况下，返回一个删除成功的信息。

如果folder下面所有的文件和文件夹都被删除成功，则删除floder，释放其资源。否则，告知用户删除失败（文件正在被使用）

1. **重命名**

重命名本身并不复杂，如果重命名文件，只需要将block中的property修改即可，如果修改的文件夹，需要将该文件夹下所有的MyDocument的fatherAddress都修改一遍。

另外需要注意的是同名的处理。

1. **读写入磁盘**

将block信息写入磁盘，由于Disk的block被固定为0，因此复原的时候只需要从block[0]入手，即可把所有的数据都恢复。

1. **搜索地址**

搜索地址，实际上就是根据contentPanel中的树形目录去查找所在的地址，然后返回即可。

1. **实现思想**

这次项目是操作系统的最后一个项目了，由于是期末，所以要写的项目也比较多。为此，这个项目也出现了一些设计上的失误，在类的设计中也都说明了这些失误。。。为此，程序没有少打补丁。

本学期操作系统的三个项目设计也都结束了，其实，虽然是以操作系统为目的的，但是，总的来说，锻炼价值还是蛮大的，比方说，这次项目就特意加强了对UI的设计（实在不想说。。我UI烂的。。），但其实还是没有达到自己预想的效果（也可能是时间太匆忙的原因把）。

由于时间有限，像复制粘贴的功能都没有做，其实，只要有了block的概念，复制粘贴并不难，只要记录下复制的是哪一个block，粘贴的时候根据那个block的信息新建一个文件就可以了，不过需要进行同名的判断。