文件系统项目文档

1252964 李贞良

目录

[设计思路 1](#_Toc344651352)

[内存模拟 1](#_Toc344651353)

[操作模拟 1](#_Toc344651354)

[缺点 2](#_Toc344651356)

[实验小结 3](#_Toc344651357)

# 设计思路

这次文件系统项目比较难。设计思路也比较复杂。

## 内存模拟

public static FloderStr[] floderCol = new FloderStr[100]; //FCB数组

public static int[] spaces = new int[100]; //位图与FAT表合一： -1　not used; 0-99 next block; 100 end;

public static String[] block = new String[100]; //物理块，与spaces数组一一对应

//当前文件夹在FCB数组中的编号这几个数据项是我用来模拟内存分配的主要工具。 floderCol 是用来表示FCB的数组。Spaces是结合了位图和FAT表的数组。 -1 表示没有使用，代表位图。 其他值的话则是代表下一个块的块号。如果是100的话代表是文件的结束块。 Block 这个块顾名思义就知道是模拟物理块的数组。

FloderStr是我自己定义的一个类。用来模拟文件或文件夹的特性。

## 操作模拟

由于这次需要将数据保存到本地文件中，所以界面设计时和以往不同，所有的空间都不能直接先建好，得重新用函数建立。我开始时就是按照老方法写的界面，所以addActionListene（）函数全部重写了。拿一个方法为例：

public static JButton getFolderCreate() {

if (FileManager.FolderCreate == null)

{

FileManager.FolderCreate = new JButton("新建文件夹");

FileManager.FolderCreate.setBounds(new Rectangle(0, 6,100,20));

FileManager.FolderCreate.addActionListener(new java.awt.event.ActionListener()

{

public void actionPerformed(java.awt.event.ActionEvent e)

{

String name=JOptionPane.showInputDialog(null,"请输入文件夹名：");

if (name.equals(""))

{

JOptionPane.showMessageDialog(null,"名字不能为空！");

}

else

{

if (!ifSameName(name,"floder"))

createFloder(name);

else

JOptionPane.showMessageDialog(null,"当前文件夹下已有同名文件夹！");

}

}

});

}

return FileManager.FolderCreate;

}

这样我的控件都是在函数里新建并返回值的，这样从本地读取时就只要重写调用这些函数就可以了。

# 缺点

这次项目相对于其他同学的来说比较粗劣。别的同学的可以响应右键点击来实现操作，和WINDOWS的界面已经很像了，而我这个显得和前两个项目的界面没什么区别。另外，我这次在内存设计上也出现了一下差错。比如只要创立文件就会分配1000的内存，没有做到合理利用。 另外，在读入系统和读出系统时也有些小BUG。

而且相较于真正的文件管理系统来说最大的缺点在于不能手动修改存放的路径，而出于时间和能力的关系自己没有解决这个问题，存在改进的空间。

# 实验小结

通过这次项目，我又学到了不少的知识。比如说在构造界面上使用一种全新的设计方法，即通过调用某个函数来add一个对象，这样看起来比较清晰，改起来也比较方便，这让我意识到仅仅在课堂上听老师讲课时远远不够的，还要通过实践来巩固学过的知识。这次我在JAVA控件的使用上遭遇了很多问题，很多控件我都无法快速查找到其API的说明，但经过一番努力后还是找到了使用指南，为以后积累了不少的经验。希望下次的项目可以完成的好一些。