**文件系统管理项目说明文档**

**一、项目说明**

1. **基本任务**

在内存中开辟一个空间作为文件存储器，在其上实现一个简单的文件系统

退出这个文件系统时，需要该文件系统的内容保存到磁盘上，以便下次可以将其回复到内存中来

1. **具体要求**

* 文件存储空间管理可采取链接结构（如FAT文件系统中的显式链接等）或

者其他学过的方法

* 空闲空间管理可采用位图或者其他方法

文件目录采用多级目录结构，目录项目中应包含：文件名、物理地址、长

度等信息

* 文件系统提供的操作：格式化，创建子目录，删除子目录，显示目录，

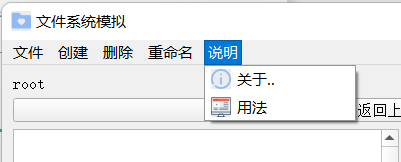
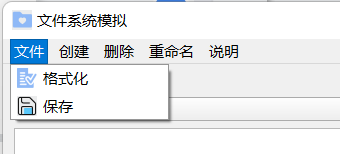
更改当前目录，创建文件，打开文件，关闭文件，写文件，读文件，删除

文件等操作

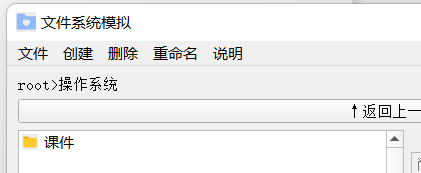
**二、功能实现**

**1. 各部分组件说明**

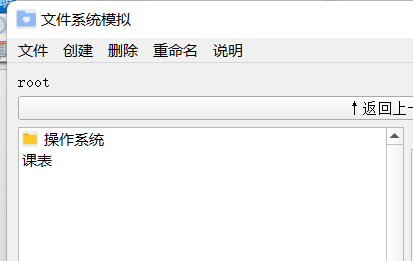
* 顶部菜单栏，提供了所有文件操作，包括格式化，保存，增加/删除/重命名和关于，教程



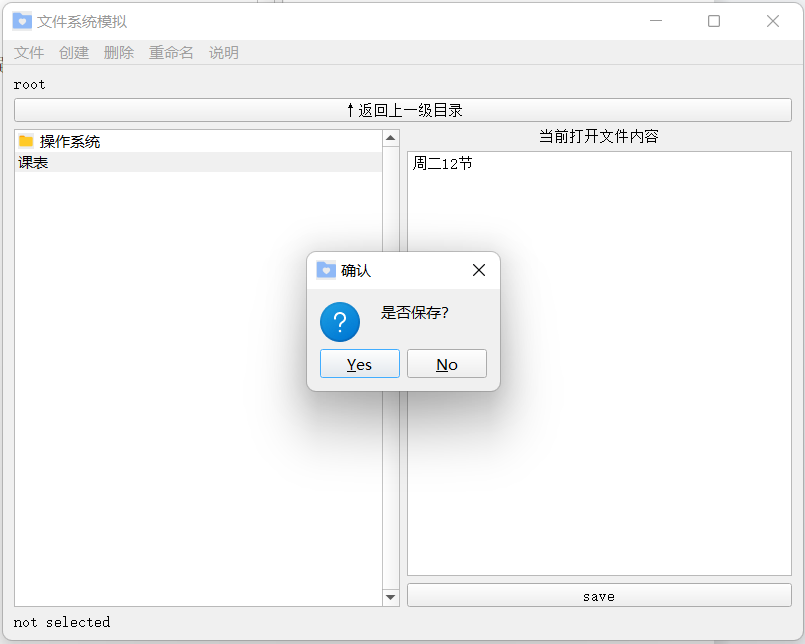
* 上方地址栏显示当前所在文件夹绝对路径



* 左侧文件目录，以列表展示当前路径下的文件和文件夹



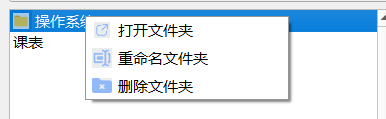
* 右侧文本编辑框，实时显示选中文件的内容，修改后点击下方 SAVE 按钮可保存



* 下方脚注显示当前选中文件/文件夹的信息



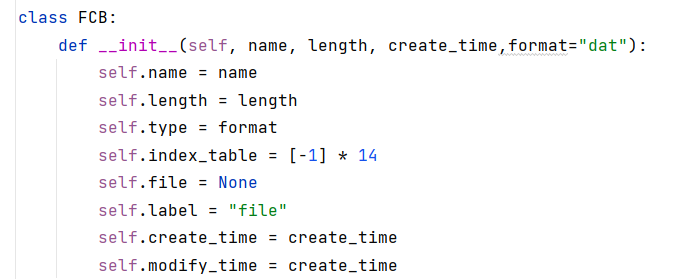
* 右键显示菜单



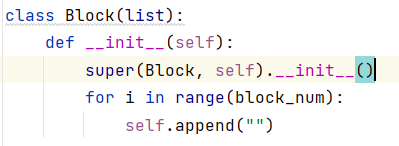
**2.实现原理**

（1）构建一个 FileSystem 类（见 file\_system.py ，可以实现文件系统的各种操作逻辑，其中：

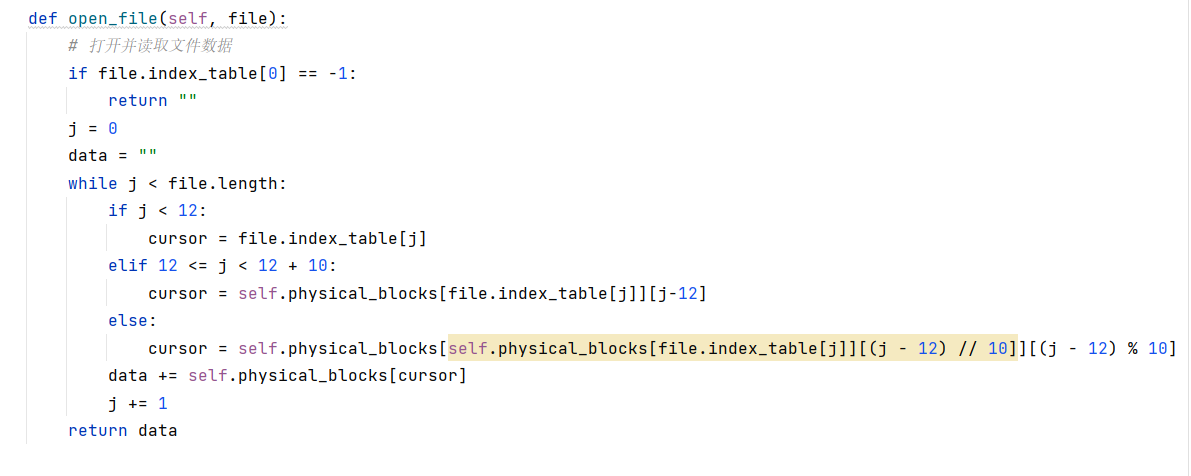
* 文件块用 FCB 类表示，存储其名称、长度、创建时间、修改时间



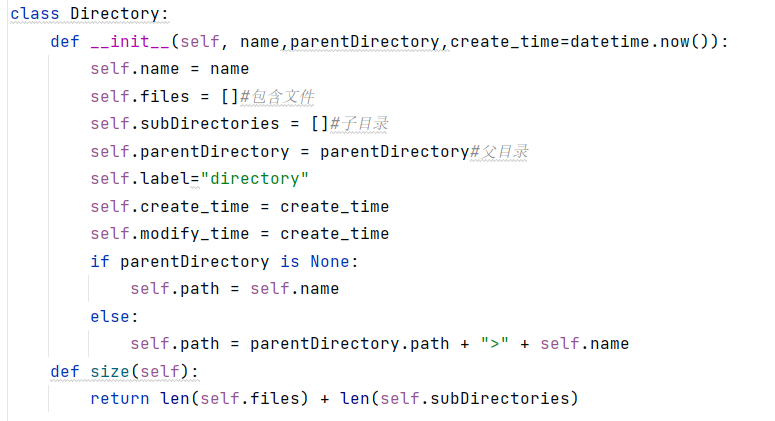
* 磁盘存储用 Block类表示



* 文件的物理结构采用索引表组织中的综合模式。代码中为每一个 FCB 类构建了索引表，以存储文件的索引项，为了简化操作，这里的综合模式与unix中的有所不同，这里每个文件索引表为12个索引项，前10项直接登记存放文件信息的物理块号（直接寻址）；如果文件大于10块，则利用第11项指向一个物理块，该块中最多可放10个文件物理块的块号（一次间接寻址）；对于更大的文件则利用第12项作为二次间接寻址。



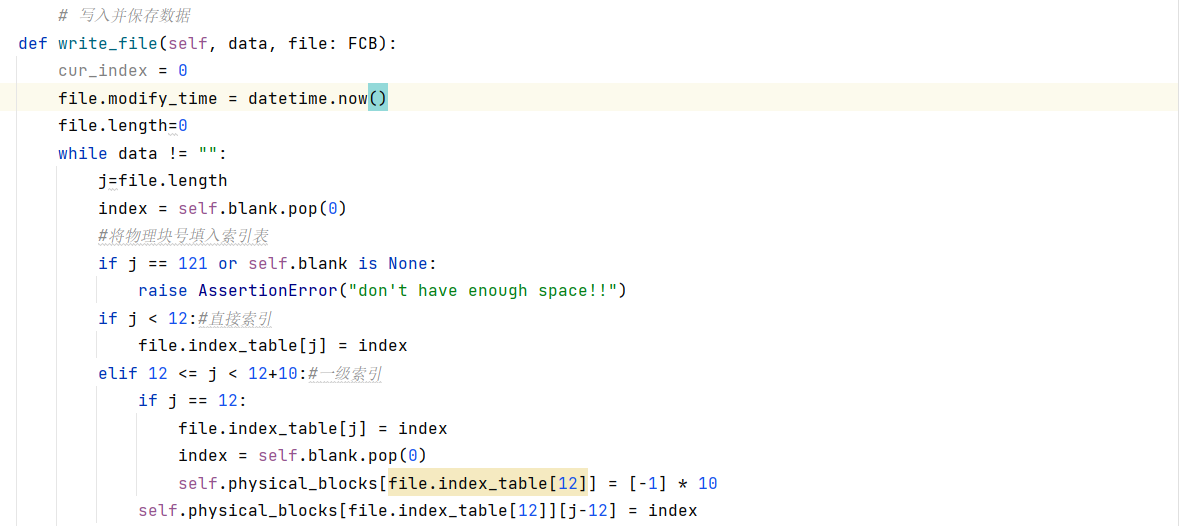
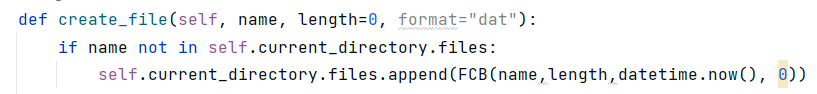
* 空闲区间用空闲块表存储，在FileSystem 类中以blank列表表示
* 多级目录通过 Directory 类中的files列表和subDirectory列表表示,它们分别存储一个文件夹中的文件和子文件夹



2

1. 文件系统主要操作的逻辑实现

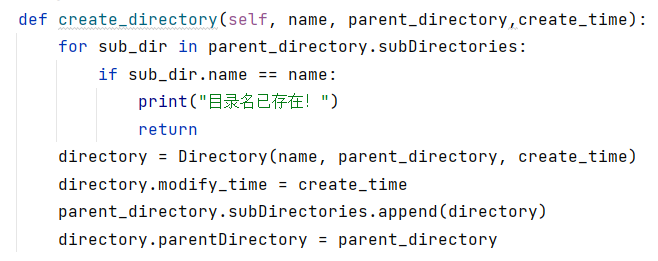
* 增加文件时，增加 FCB 块，并加入到相应Directory中；写文件时，额外写入到磁盘 Block中，同时从blank列表中删除被占用的物理块号：



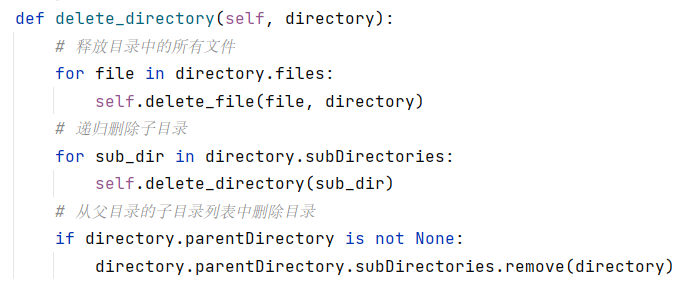
* 删除文件同理：



* 新建文件夹时，增加 Directory块，并加入到当前文件夹的子目录列表subDirectory中：



* 删除文件夹需要将其中的文件逐一释放并且递归删除其子文件夹：



1. 文件系统UI界面实现

借助PyQt构建对应的UI界面，获取鼠标事件信息，判断点击项并触发相应操作，每次操作后更新界面

**三、运行方法**

1.**直接运行**

已经通过 pyinstaller生成了Windows上的可执行文件。在Windows系统上点击 file\_system.exe ，即可运行程序

2.**编译运行**

Python version：3.8

第三方库：PyQt5，qt-material（用来应用本程序所需的UI样式）

运行文件：file\_system\_ui.py

1. **运行截图**

