# 设计方案报告

## 开发环境

项目采用C++编程语言开发，使用Microsoft Visual Studio Community 2017进行开发。

## 项目功能

本项目实现了一个文件系统，并且在退出文件系统时，将文件系统的内容保存，在下次将其数据恢复。

文件系统主要实现的功能有：（相关指令）

1. 格式化---format
2. 创建子目录---cdir
3. 删除子目录---ddir
4. 显示目录---ls
5. 更改当前目录---cd
6. 创建文件---create
7. 打开文件---open
8. 关闭文件---close
9. 写文件---write
10. 读文件---read
11. 删除文件---delete
12. 回到父目录---back
13. 存储文件系统数据
14. 读取文件系统数据

## 算法：

本项目对于空闲空间管理采取位示图管理，文件目录采取多级目录结构，文件存储空间管理采用索引表的方式管理。

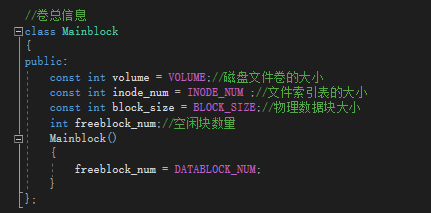
对于空闲空间，建立vector<int>inodebitmap进行管理；

对于文件系统内容的保存，在系统退出时，将文件位示图和块位示图写入block.txt中，再依次写入文件、目录的相关信息；同时，将文件内写入的数据，统一按块写入data.txt中存储。在文件系统再次打开时，文件系统将自动读取文件系统数据，恢复到之前的状态。

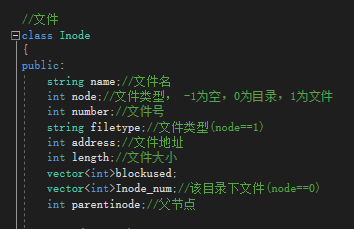
## 总体设计

**本项目设计的数据结构有**

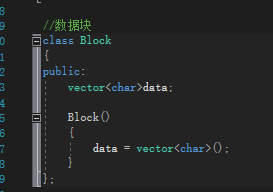
1. **卷总信息**

记录了系统缓存的各项信息，包括磁盘文件卷的大小，文件索引表的大小，物理数据块的大小以及空闲块的数量。

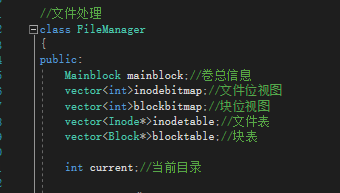
1. **文件信息**

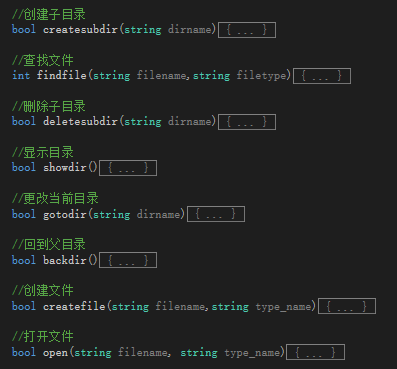
记录了文件的索引号，文件名称，文件类型，文件地址，文件大小，文件数据块索引，目录结点下的文件，父节点等信息。

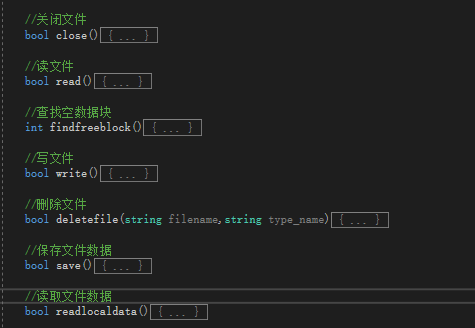
1. **数据块**

记录一个块存储的信息

1. **文件处理**

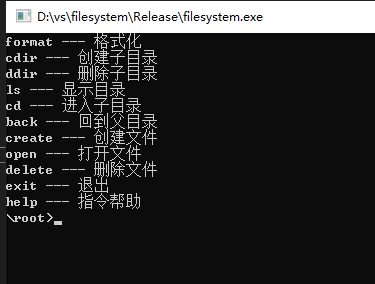
记录了当先卷总信息，文件位示图，块位示图，文件表，块表，当前目录。

**本项目主要调用的函数有以下几个：**

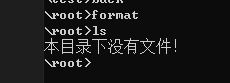


## 用户界面：

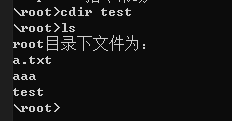
**初始界面：**

****

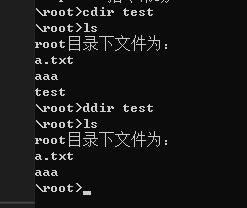
**格式化---format**

****

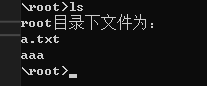
**创建子目录---cdir**

****

**删除子目录---ddir**

****

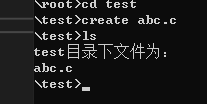
**显示目录---ls**

****

**更改当前目录---cd**

****

**创建文件---create**

****

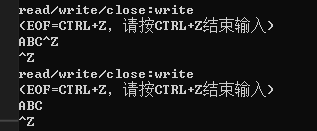
**打开文件---open**

****

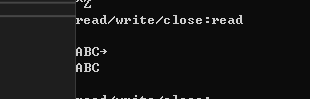
**关闭文件---close**

****

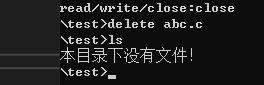
**写文件---write**

****

**读文件---read**

****

**删除文件---delete**

****

**回到父目录---back**

****