**操作系统课程设计任务书**

**一、课程设计任务**

本次课程设计的任务是虚拟实现部分操作系统的典型算法，加深对操作系统运行机制的掌握和理解。本部分任务每组同学完成一份（分组见“课程设计分组”部分）。任务具体要求：

在Winidows/Linux/iOS/Android平台下（平台任选一个），以面向对象思想实现下面功能：

1、磁盘管理

建立一个4KB大小的文件模拟磁盘，按逻辑将其划分为1024块，每块大小4B。其中900块用于存放普通数据，124块用于存储兑换数据。存储管理需要支持：

（1）数据组织：对需要存放的文件数据加以组织管理，可以采用连续组织方式、显式连接（FAT）方式、单级索引组织方式、二级索引组织方式、混合索引方式（每组要求不同，具体见“课程设计分组”部分，下同）。

（2）空闲块管理：能够查询并返回当前剩余的空闲块，对空闲块管理可以采用位示图法、空闲盘块表法、空闲盘块连法、成组连接法。

（3）兑换区管理：能够写入、读出兑换区数据。

2、目录管理

为写入模拟磁盘的数据文件建立目录，目录可以是单级文件目录、双级文件目录、树形结构目录。在目录中选择某个文件可以将其数据读入模拟内存。目录中包含文件名、文件所有者、创建时间、文件结构、在磁盘中存放的地址等信息。目录管理需要支持：

（1）新建目录：在目录中新建空目录

（2）删除目录：删除空目录

（3）为文件建立目录项：一个文件被创建后，为该文件创建目录项，并将文件相关信息写入目录中。

（4）删除文件：删除目录中某个文件，删除其在磁盘中的数据，并删除目录项。如果被删除文件已经读入内存应该阻止删除，完成基本的文件保护。

3、内存管理

申请一块64字节的内存空间模拟内存，按逻辑划分为16块，每块4B。将目录中选中的文件读入内存，显示文件中信息。内存可以同时显示多个文件信息，每个文件固定分配4个内存块，如果4个内存块不能显示文件全部信息，采用页面置换策略，将已显示完的页换出内存，可以选择的置换策略有，全局置换、局部置换、FIFO、LRU。内存管理需要支持：

（1）分配内存块：为线程分配内存块，每个线程默认分配4块。

（2）回收内存：线程结束后回收其内存。

（3）空闲内存块管理：为进入内存的数据寻找空闲内存块。没有空闲内存时，应给出提示。

（4）块时间管理：提供数据块进入模拟内存的时间、访问时间的记录和查询功能，为页面置换算法提供支持，当某个内存块被选中时，更新其访问时间。

4、线程管理

本虚拟系统以线程为基本运行单位，线程本身采用编程语言提供的线程机制，不模拟。系统主要包括的线程有：·

（1）数据生成线程：该线程负责生成外存数据，给定数据大小（按字节计算）、数据信息（英文字母）、存储目录、文件名后，该线程调用磁盘管理中空闲磁盘管理功能，申请所需大小的外存块，如果盘块不够给出提示。按照要求的数据组织方式，将数据存入磁盘块（按块分配磁盘），并调用目录管理功能为其在目录中建立目录项，更改空闲盘块信息。双线程 非并行

（2）删除数据线程：该线程调用删除目录管理中文件删除功能删除数据（内存中文件不能删除）。并回收外存空间，更新空闲盘块信息。双线程

（3）执行线程：选择目录中的文件，执行线程将文件数据从外存调入内存，为此，首先需要调用内存管理的空闲空间管理功能，为该进程申请4块空闲内存，如果没有足够内存则给出提示，然后根据目录中文件存储信息将文件数据从外存读入内存，此间如果4块内存不够存放文件信息，需要进行换页（选择的换页策略见分组要求），换出的页面存放到磁盘兑换区。允许同时运行多个执行线程。文件数据在内存块的分布通过线程的页表（模拟）进行记录。

（4）线程互斥：对于64B的内存，线程需要互斥访问，避免产生死锁。不能访问内存的线程阻塞，等待被唤醒。

5、用户接口

对内存块、外存块、目录信息进行可视化显示，并能够动态刷新。文件调入内存过程、以及换页过程在块与块之间加入延时，以便观察。

对于实现以上功能，可以采用任何熟悉的编程语言，不做具体要求。