

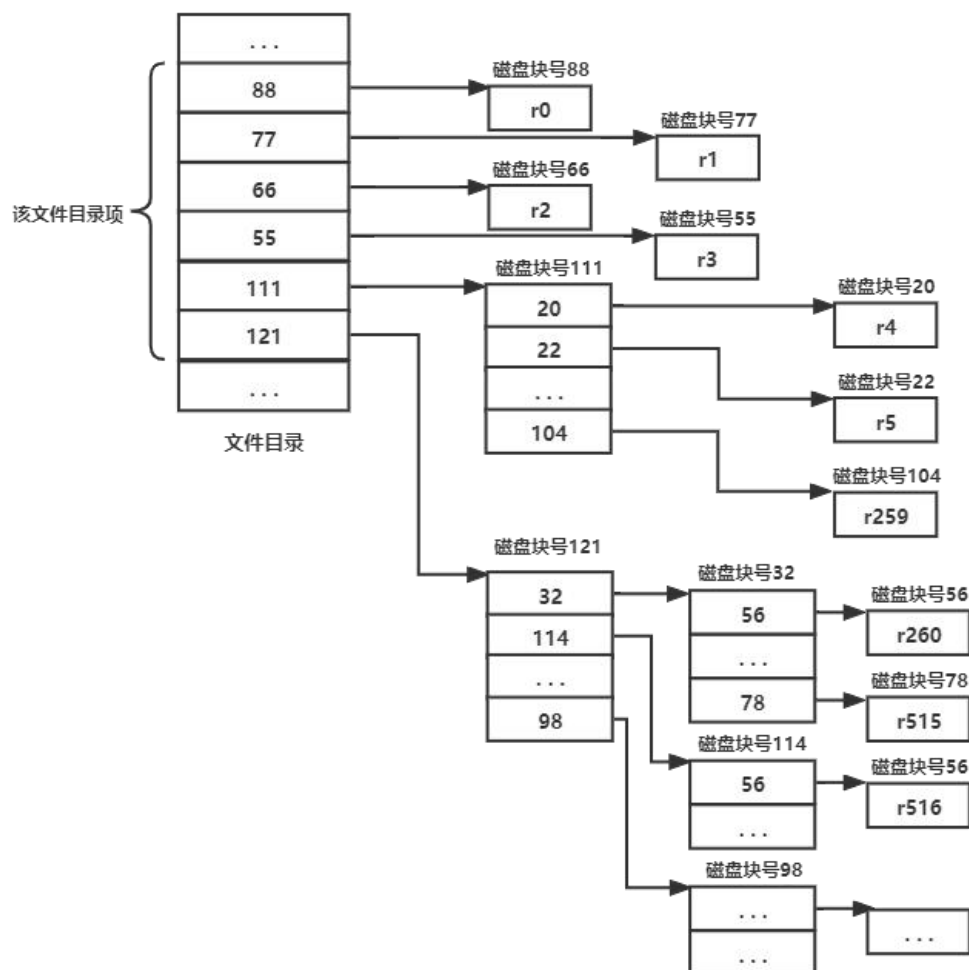
## 第 9 章作业

9-36、

(1) 采用连续文件结构读取最后 1024 字节需要 1 次读磁盘操作。最后 1024 个字节的相对块号为 99，那么 首块号+99 为读取磁盘块号，故只需读取首块号+99 的那块磁盘的 1024 个字节即可。

(2) 采用串联文件结构读取最后 1024 字节需要 101 次读磁盘操作。因为每块中还需要一个字的存放指针信息，故总共需要 101 个磁盘块。而串联文件结构需要从第一个磁盘块开始遍历，直到到达目的磁盘块才算读取到有效信息，故在读取最后 1024 个字节时，则需要将所有的磁盘块都读取一边，故需要读取 101 个磁盘块。

(3) 采用多级索引文件结构读取最后 1024 字节。文件索引结构图如下：



在文件已打开，即文件目录已读取到内存的前提下，当读取最后 1024 个字节时，经计算可知，其存放在一级索引块下的某个块中，故通过图可知，需要访问磁盘 2 次就可找到该块。一次是访问一级索引块，一次是读取数据。

### 9-37、

(1) 由题意可知, 每个目录文件最多占 4 个磁盘块, 且每个文件目录项占 102 个字节, 一个磁盘块大小为 1KB。由于是采用串联文件结构, 那么每个磁盘块还需要 4B 的空间放置下一磁盘块号。那么一个目录下最多文件数为  $4 \times (1\text{KB} - 4\text{B}) / 102\text{B} = 40$  个。故一个目录下最多有 40 个文件。

对于普通文件, 有 7 项直接索引, 可以存放 7KB 的数据; 有 2 项一级间接索引, 可以存放  $1\text{KB} / 4\text{B} \times 1\text{KB} = 256\text{KB}$ 。那么当普通文件大小大于  $7 + 256 = 263\text{KB}$  时, 系统将需要为其建立二级索引表。

(2)

#### 1. 判断最少读取次数:

当需要的目录都在第一个块中时, 则每次获取目录只需要访问一次磁盘。

打开 home 目录: 1 次, 打开 user 目录: 1 次, 打开 os 目录: 1 次, 打开 os1 目录: 1 次。故至少需要读取硬盘 4 次。

#### 2. 判断最多读取次数:

当需要的目录都在第四个块中时, 则每次获取目录需要最多访问四次磁盘。

打开 home 目录: 4 次, 打开 user 目录: 4 次, 打开 os 目录: 4 次, 打开 os1 目录: 4 次。故最多需要读取硬盘 16 次。

(3)

普通文件, 有 7 项直接索引, 可以存放 7KB 的数据; 有 2 项一级间接索引, 可以存放  $1\text{KB} / 4\text{B} \times 1\text{KB} = 256\text{KB}$ ; 有 1 项二级间接索引, 可以管理  $1\text{KB} / 4\text{B} = 256$  个一级间接索引表, 那么总共可以存放  $256 \times 256\text{KB}$  的数据。

故普通文件最大可达  $7 + 256 + 256 \times 256 = 65799\text{KB}$ 。

达到最大时, 在文件目录已打开的前提下, 当读取位于二级间接索引块下的某个字节数据时, 需要读取的磁盘块最多为 3 个。其中前两次读取索引, 最后一次读取数据。

### 9-38、

(1)

1. usr 目录在根目录的第 2 块, 而根目录下的第一块是常驻目录, 故只需访问一次硬盘获得 usr 目录块号或地址;

2. 在 usr 目录下的第一块, 访问一次硬盘获取 you 目录的块号或地址;

3. 在 you 目录下的第一块, 访问一次硬盘获取 dir1 目录块号或地址;

4. 在 dir1 目录下的第一块, 访问一次因硬盘获取 A 目录块或地址;

5. 在 A 目录下, 开始访问记录。A 包含 590 个逻辑记录, 一个块只能存放 2 个逻辑记录, 故需要 295 块

故将 A 读入主存, 至少访问  $4 + 295 = 299$  次硬盘。

(2) 那么在前 4 次访问的基础上, 已经打开了 A 目录。那么在连续结构的基础上, 要读取第 480 号, 则可直接通过计算, 算出需要访问的相对块号为 239, 则可直接访问 (首块号+239) 块的数据, 即可将第 480 号记录读到内存。此过程只读取一次硬盘。

故将 A 的第 480 号记录读入主存, 至少要访问  $4 + 1 = 5$  次硬盘。

## 9-39、

(1)

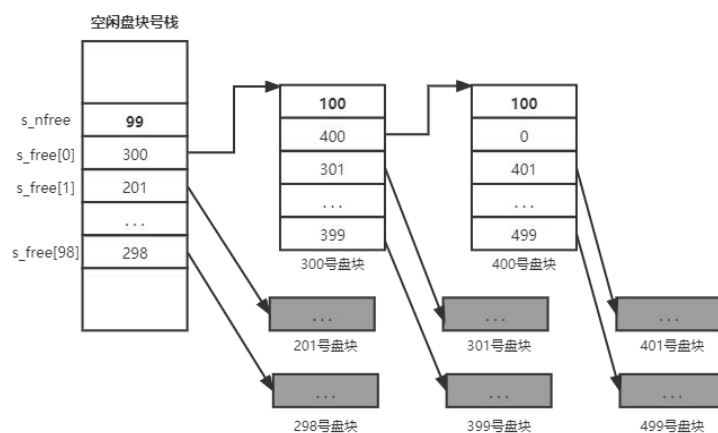
目前还剩 301 块空闲盘块。

第 4 组空闲块有 2 个，第 3 组空闲块有 100 个，第 2 组空闲块有 100 个，第 1 组空闲块有  $100-1 = 99$  个，故总的空闲块数为  $2+100+100+99 = 301$  块。

(2)

首先看空闲盘块号栈顶，将 100 号盘块分配出去。此时  $s\_nfree = 1$ ，需要将 200 号盘块的内容拷贝到空闲盘块栈中，并分配 200 号栈。此时  $s\_nfree$  变为 100。再将栈顶的块号 299 号块分配出去，此时  $s\_nfree = 99$ 。至此，完成了 3 个磁盘块的分配，依次为 100、200 和 299。

分配完的空闲存储空间为：



(3) 在上图的基础上，删除另一个文件并回收 5 个盘块：700、711、703、788、701。

在回收 700 时， $s\_nfree = 99$ ，将 700 块号放置  $s\_free[99]$  的位置。 $s\_nfree = 100$ ，然后在回收 711 块号的块时，此时栈满，将  $s\_free$  和  $s\_nfree$  放入释放块 700 中，将 700 块号放入  $s\_free[0]$ ， $s\_nfree$  置 1。那么在回收 703、788 和 701 时，只需如回收 700 那样回收即可。回收后的盘块链接情况如下图。

