

《操作系统》实验报告

实验名称 文件管理

学 院 信息科学与技术

专业班级 计算机科学与技术

小组成员 王倩芸 何茂林 段倩薇 王昊笛 袁玥 戚镇麒 王驰翔

2016 年 12 月 10 日

一、【实验目的】

通过阅读和调试文件管理的模拟程序以加深对文件系统的理解。

二、【实验任务】

1. 每个小组给出该模拟文件系统中的各个模块的流程图及其各模块之间关系的说明（电子版）；
2. 调试完善模拟程序。

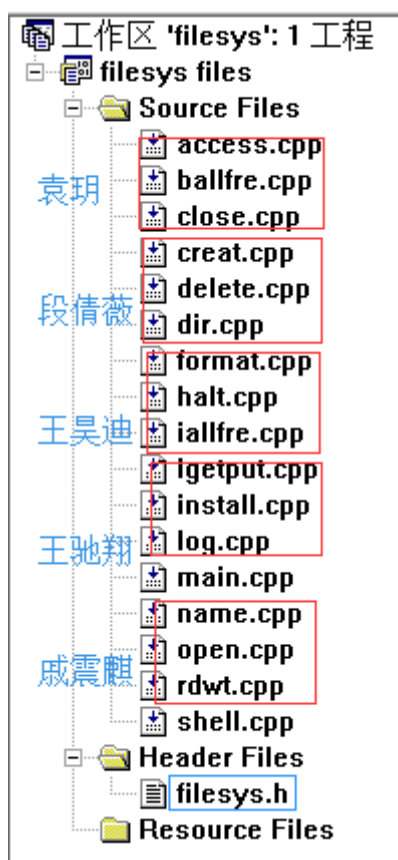
三、【实验过程】

1. 小组分工-制作各个模块的流程图

i. 分工安排：

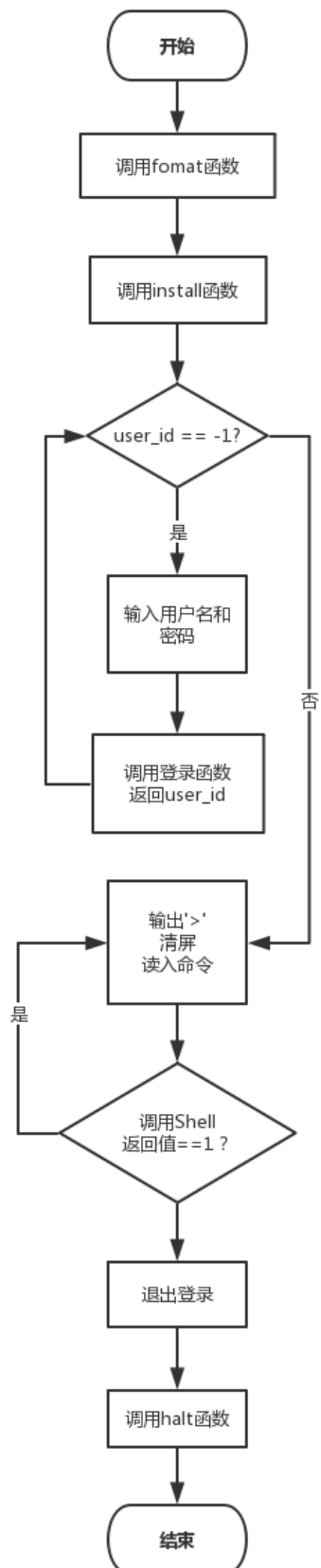
何茂林与王倩芸制作 Main.c 和 Shell.c 流程图，并调试两个 BUG。

剩下同学分工如下：

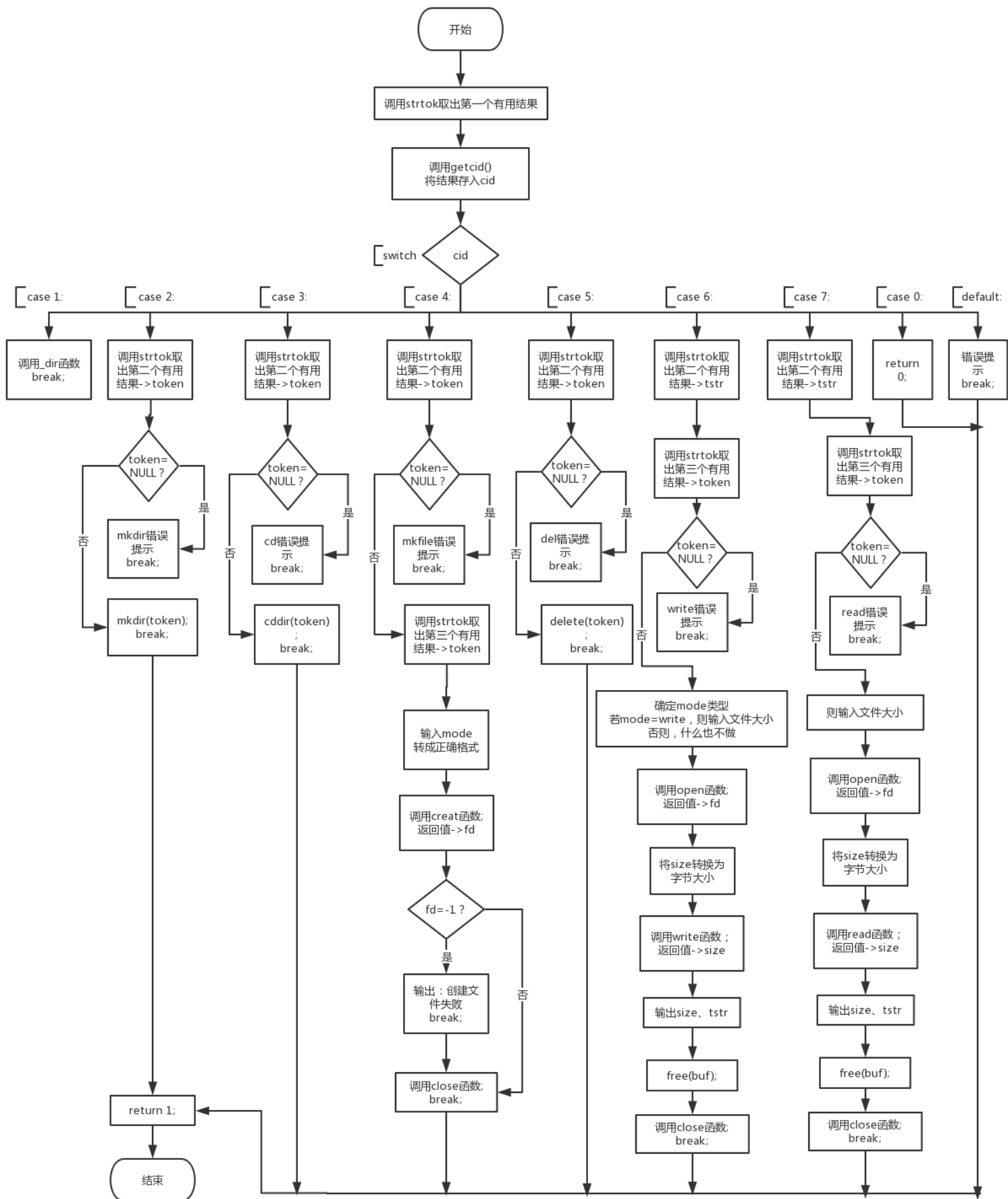


ii. 流程图展示:

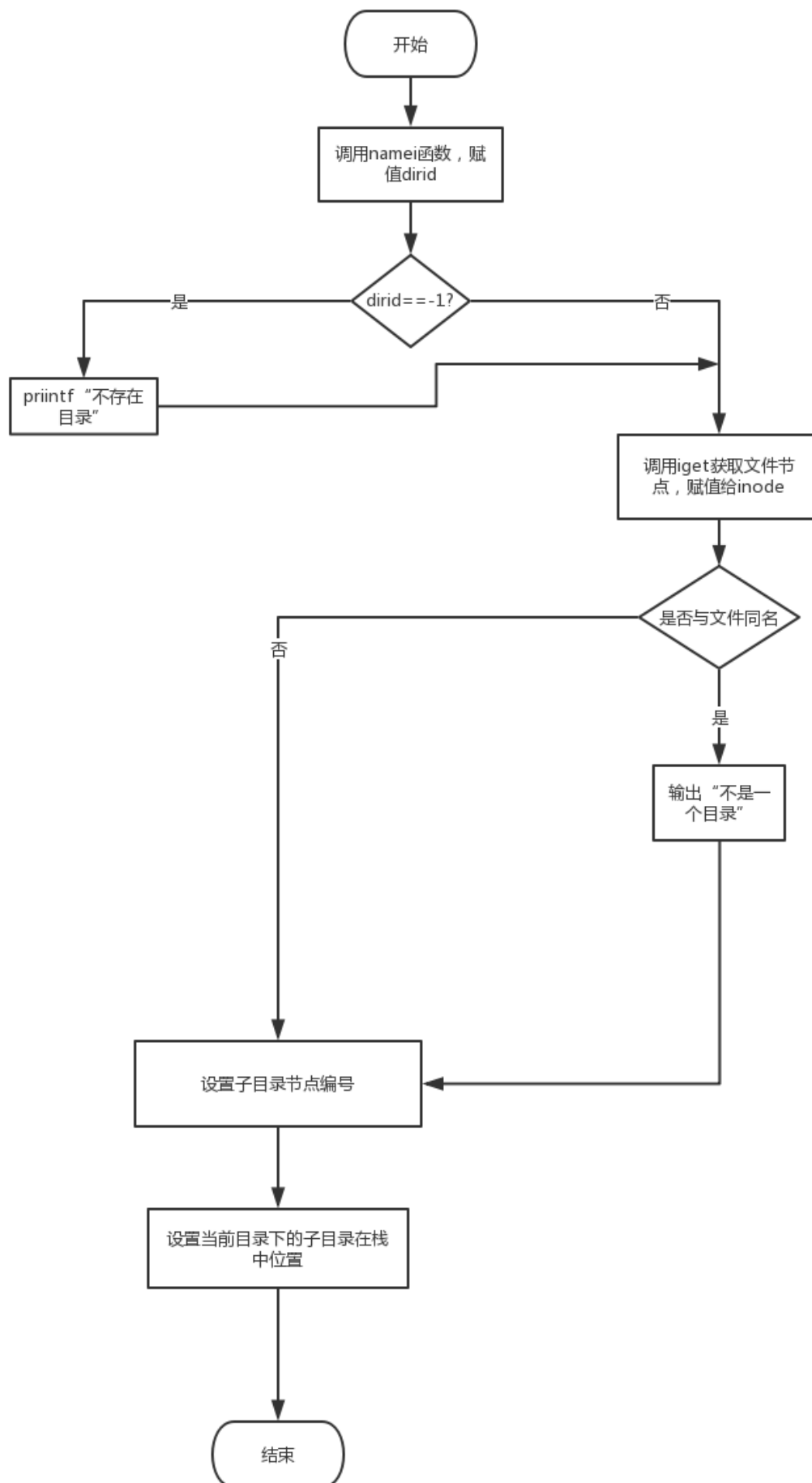
- 1、 Main 函数：提供程序入口和登陆界面，调用 Shell 函数和 Login 函数，实现功能。最后调用 halt 函数，收尾。



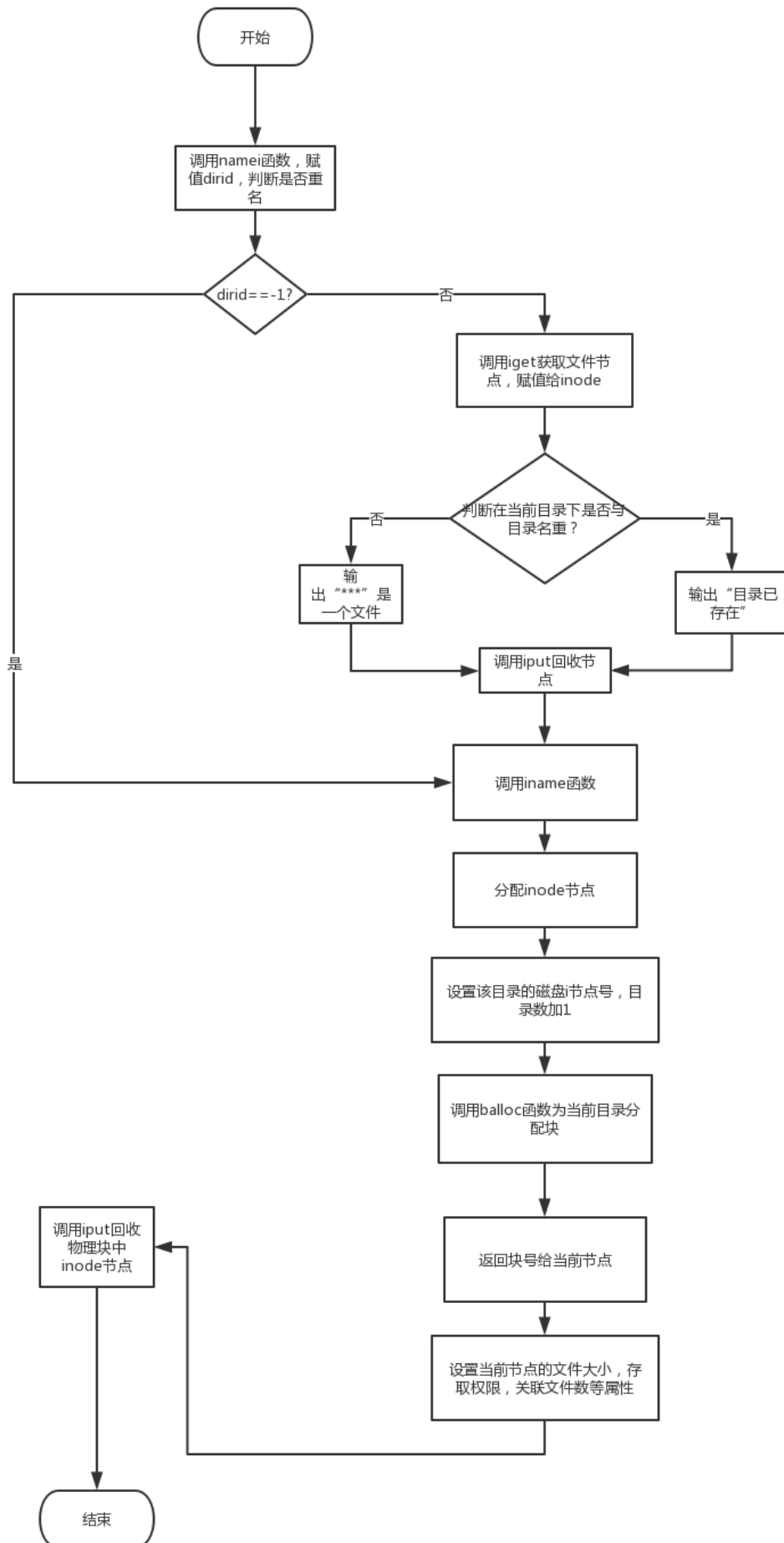
2、 Shell 函数：处理不同类型的命令，分别调用不同函数。



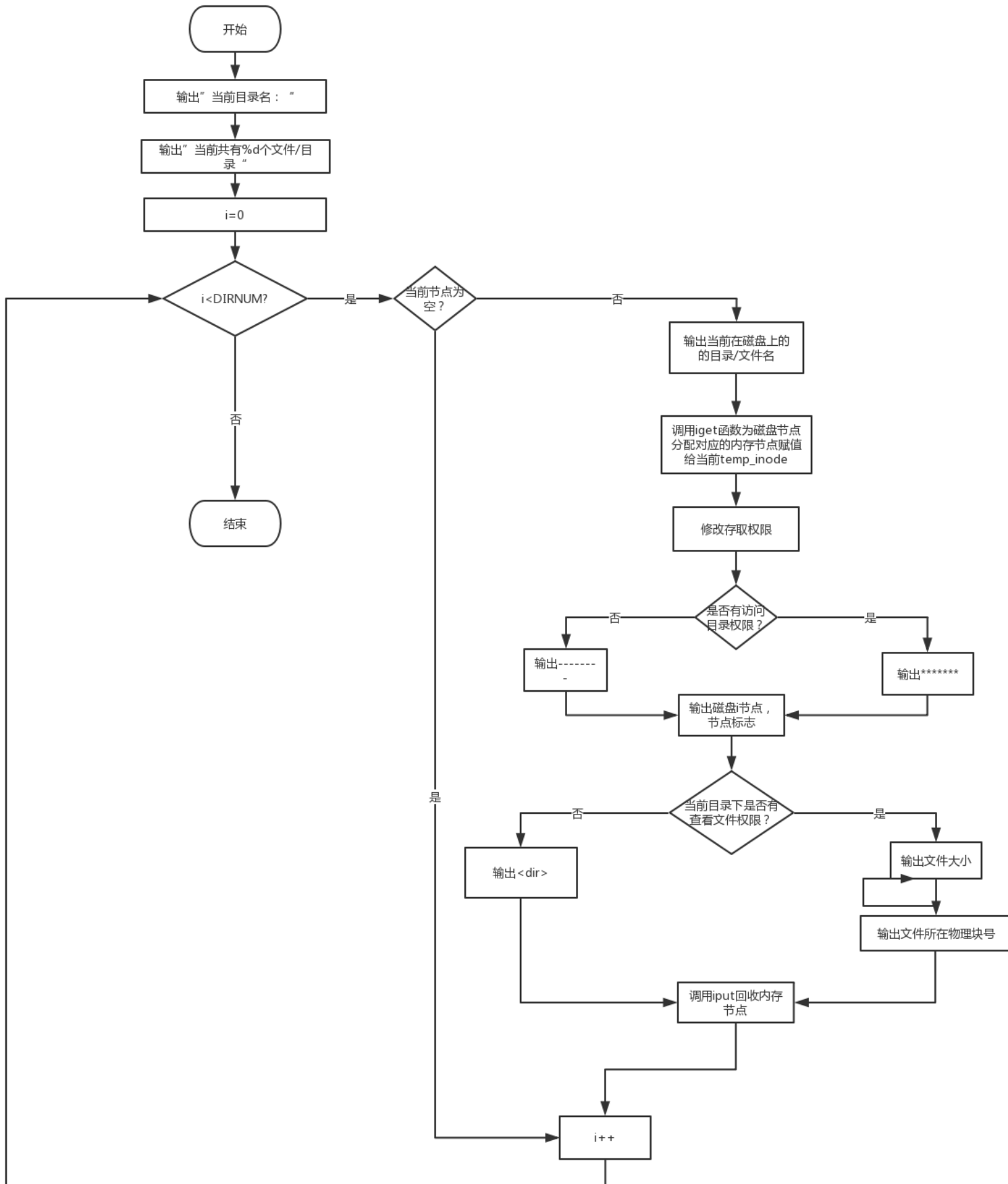
3、 chdir 函数：实现 cd 命令功能。



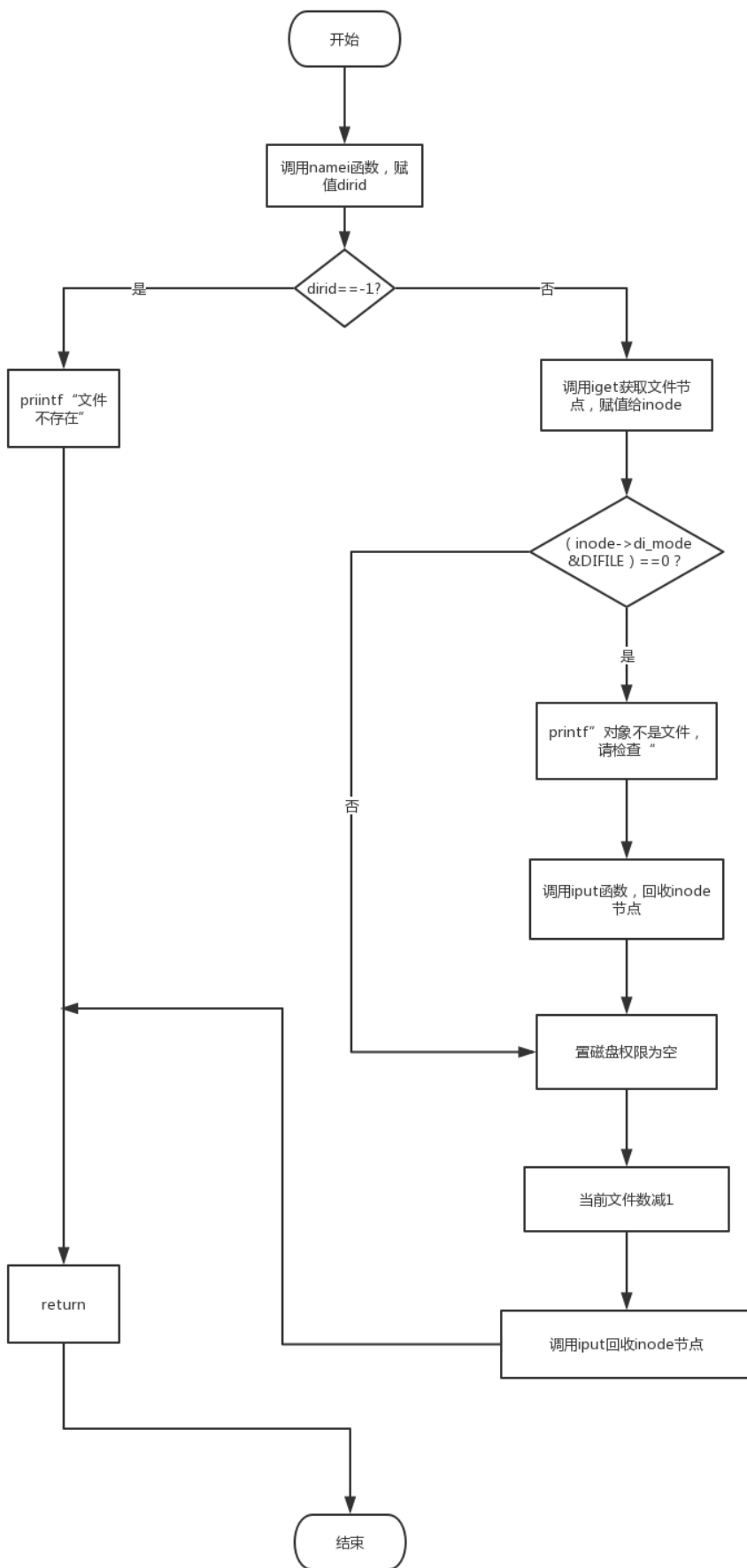
4、 mkdir 函数：实现命令 mkfile。



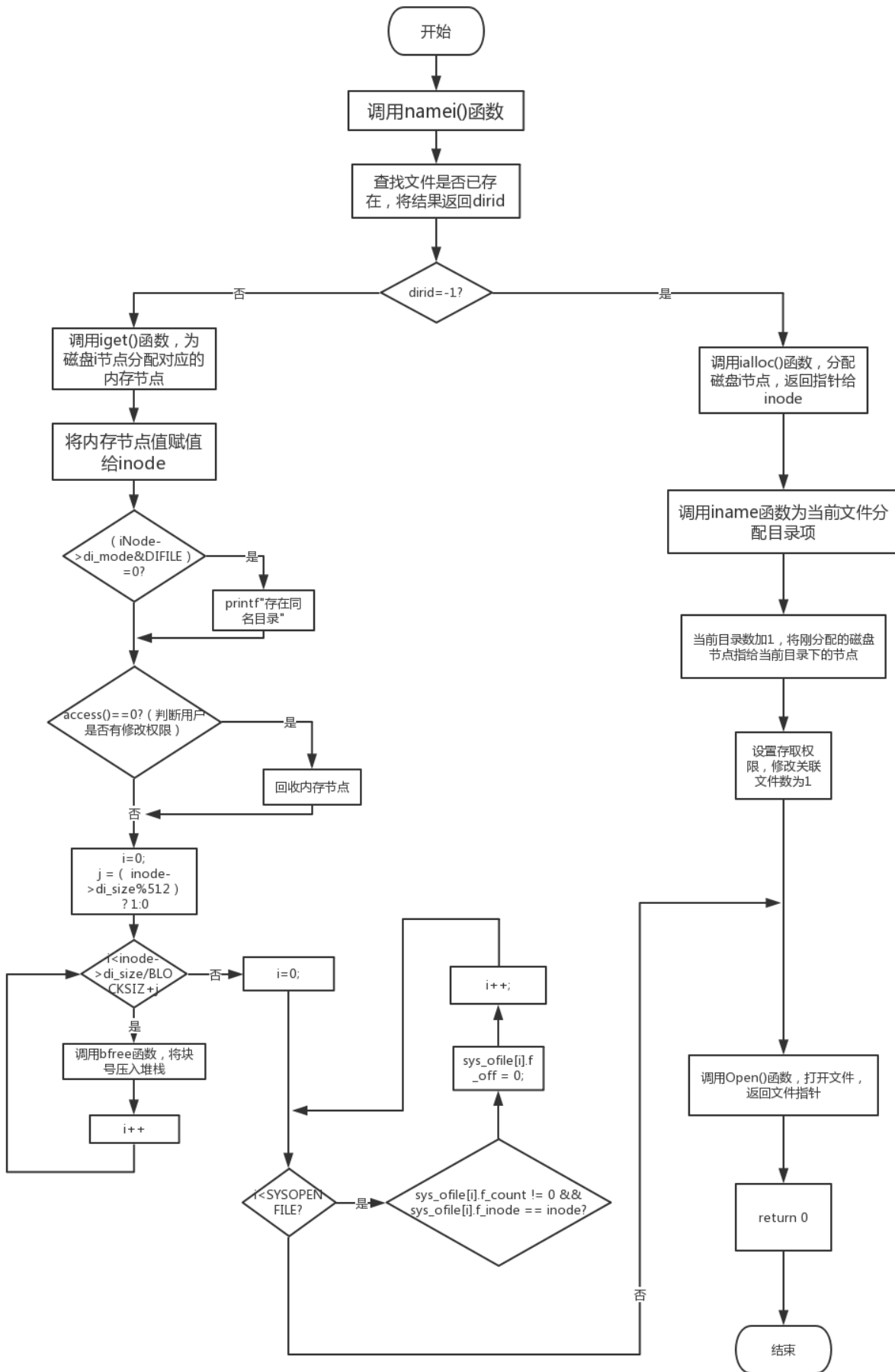
5、 _dir 函数：实现命令 dir。



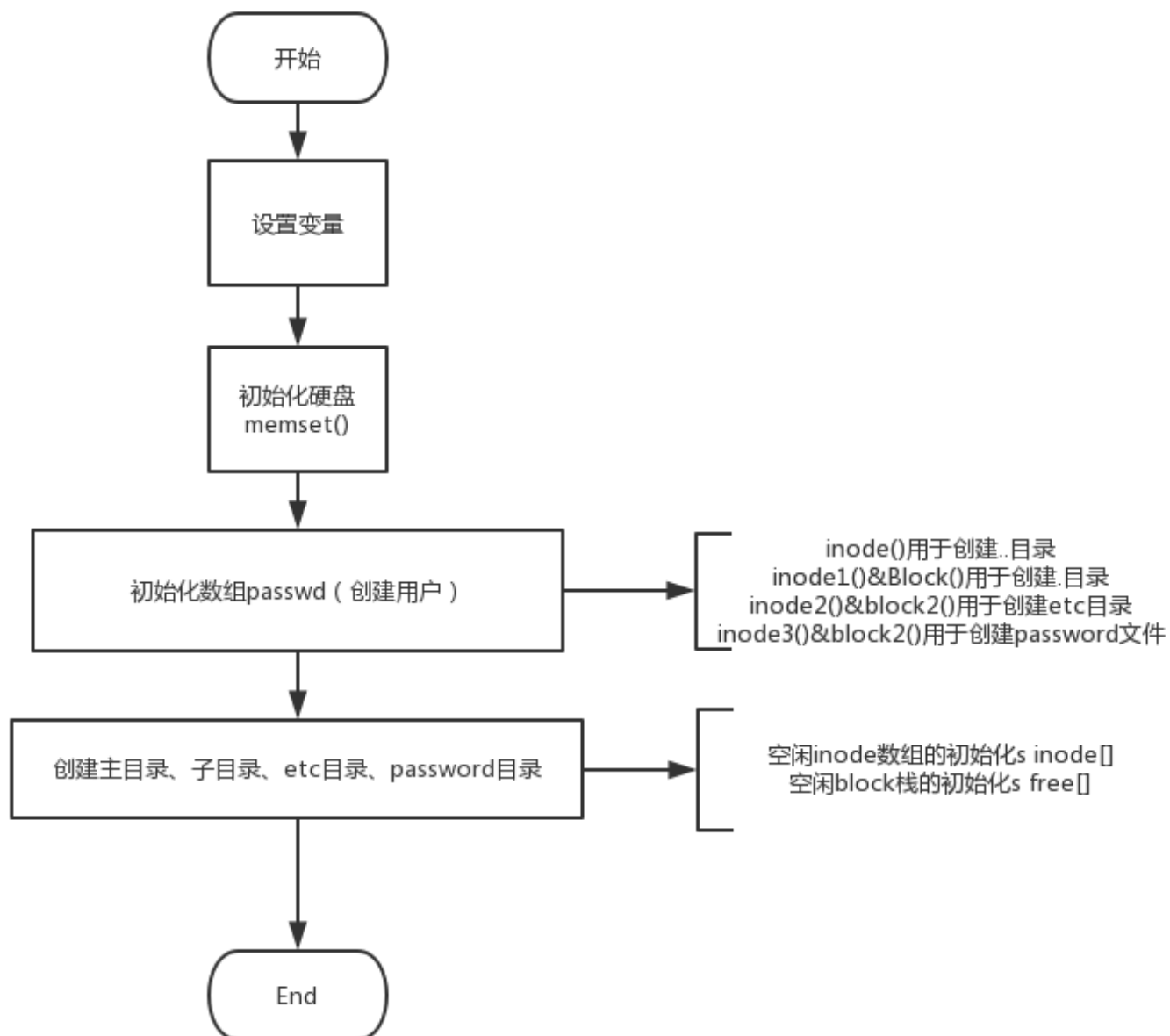
6、 delete 函数：实现命令 del。



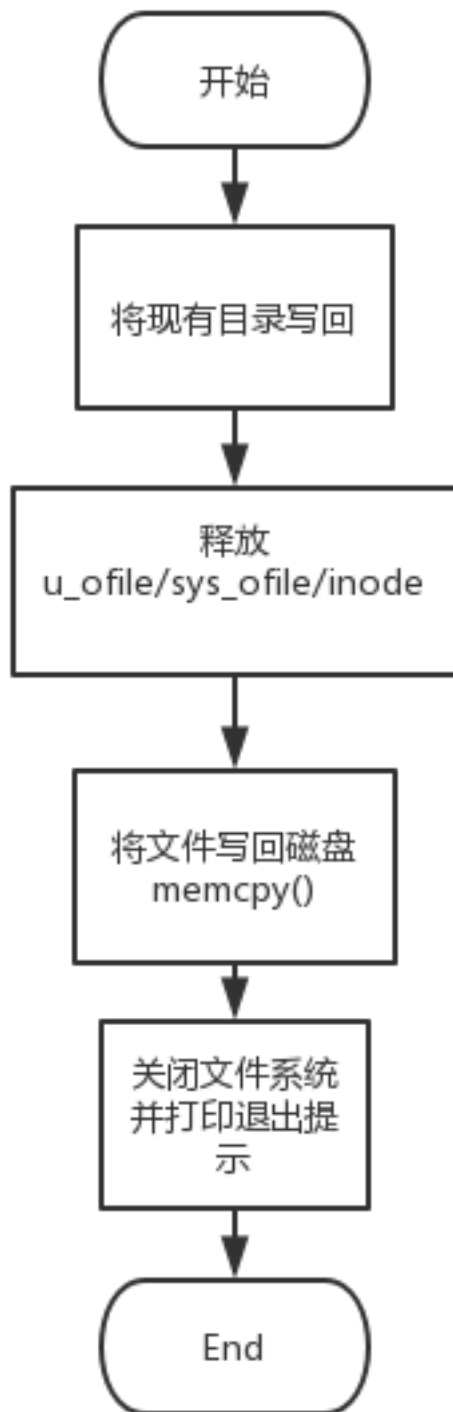
7、 create 函数：创建文件。



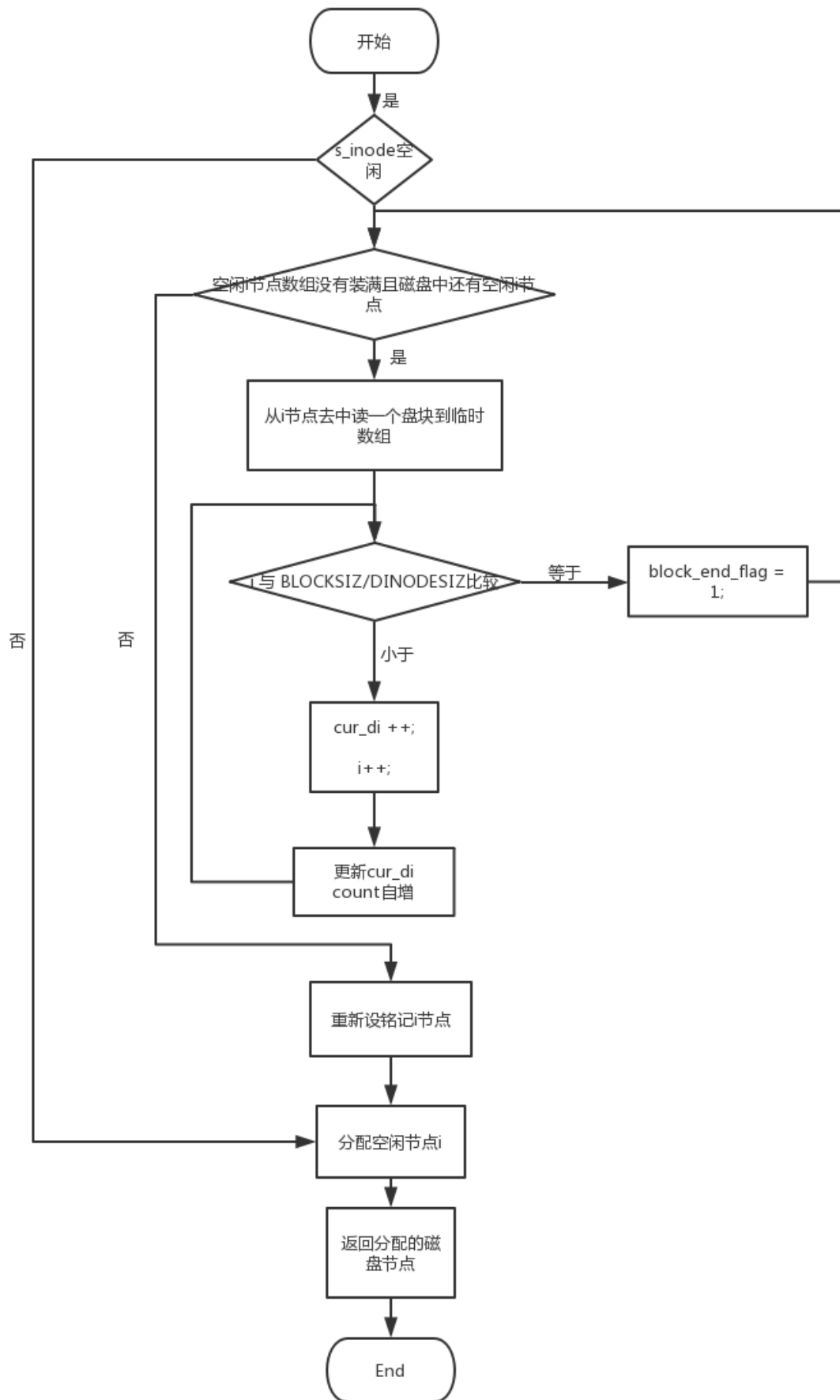
8、 format 函数：初始化硬盘、password 数组，创建目录。



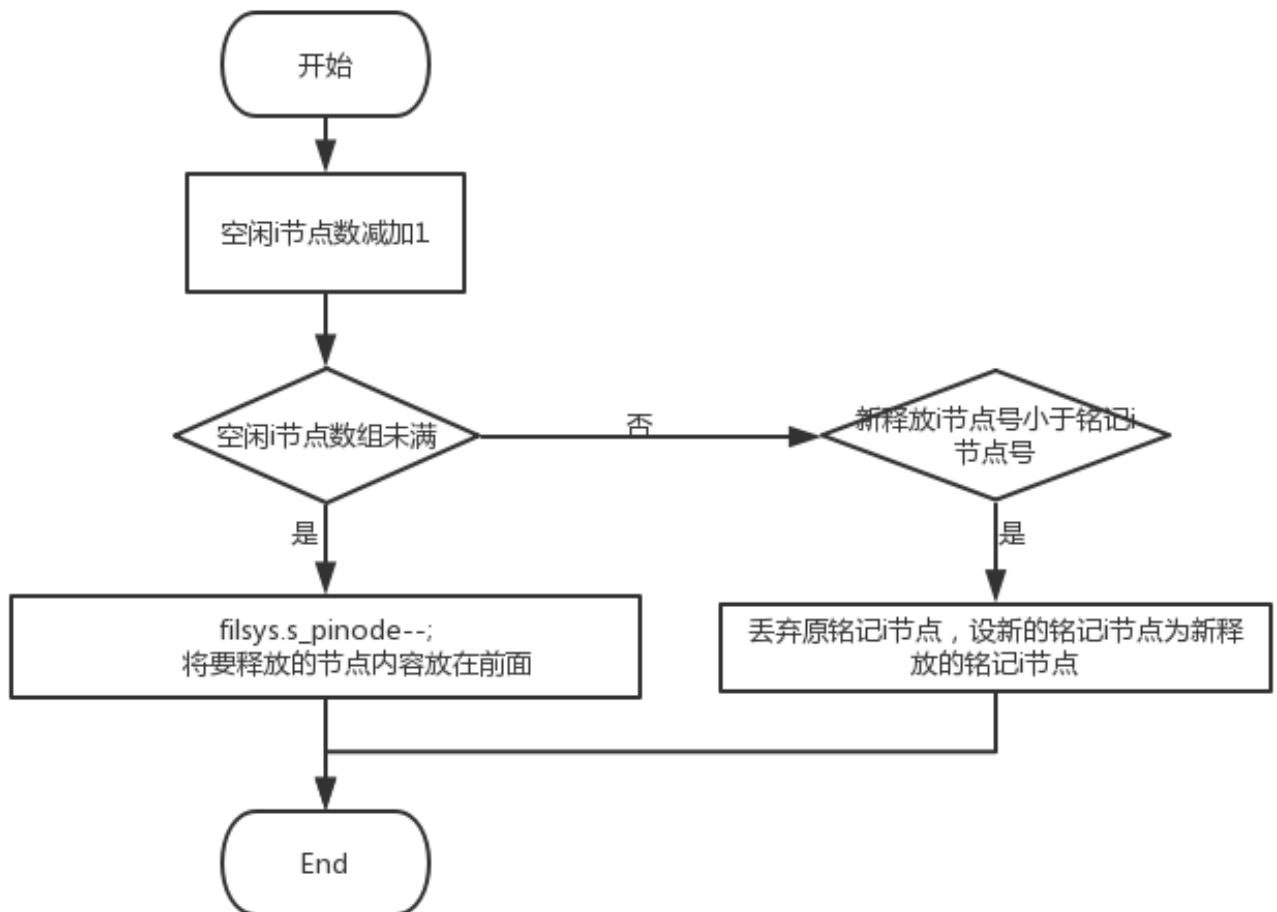
9、 halt 函数：写回磁盘，收尾。



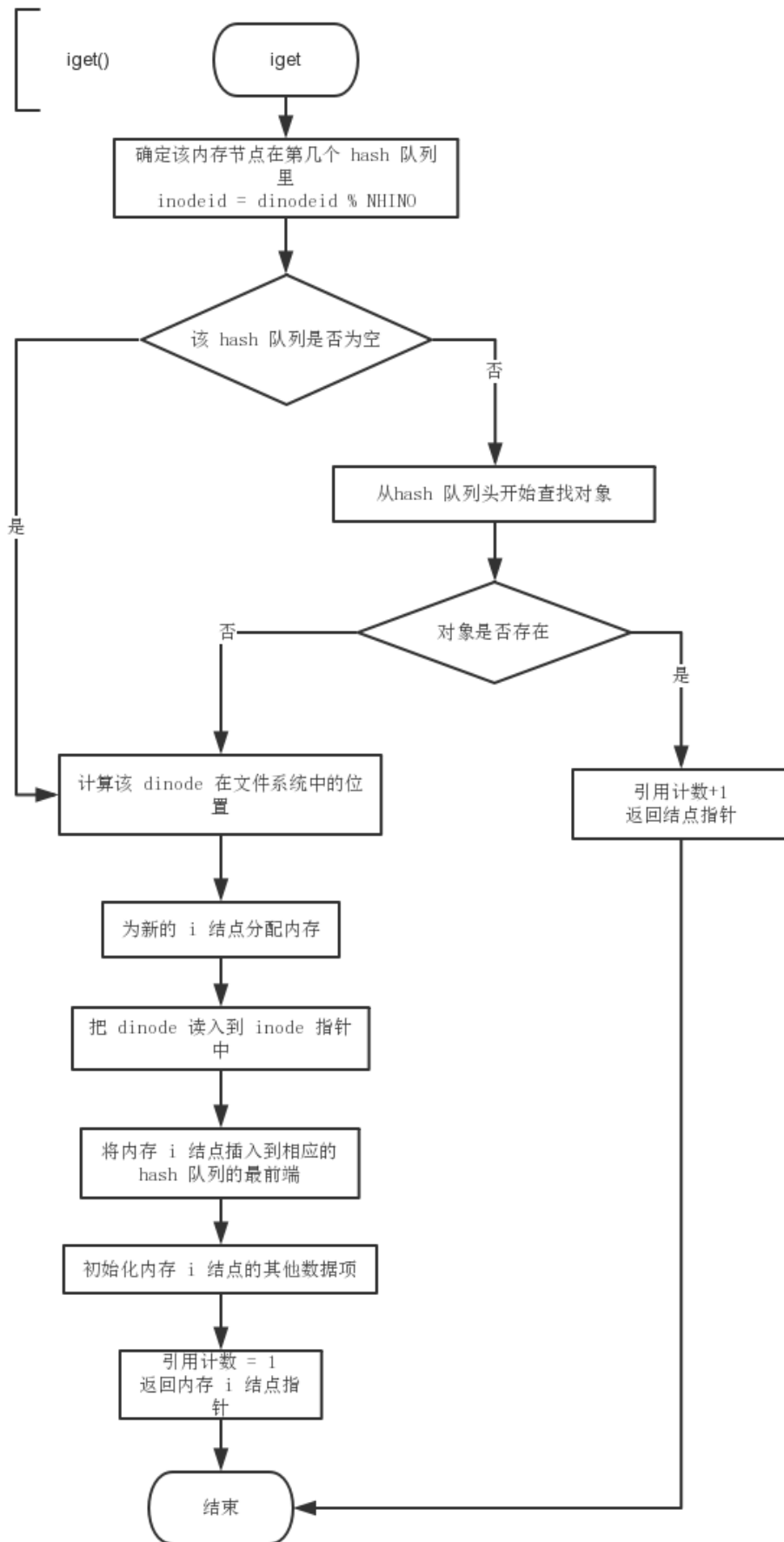
10、 iallc 函数：分配盘块。



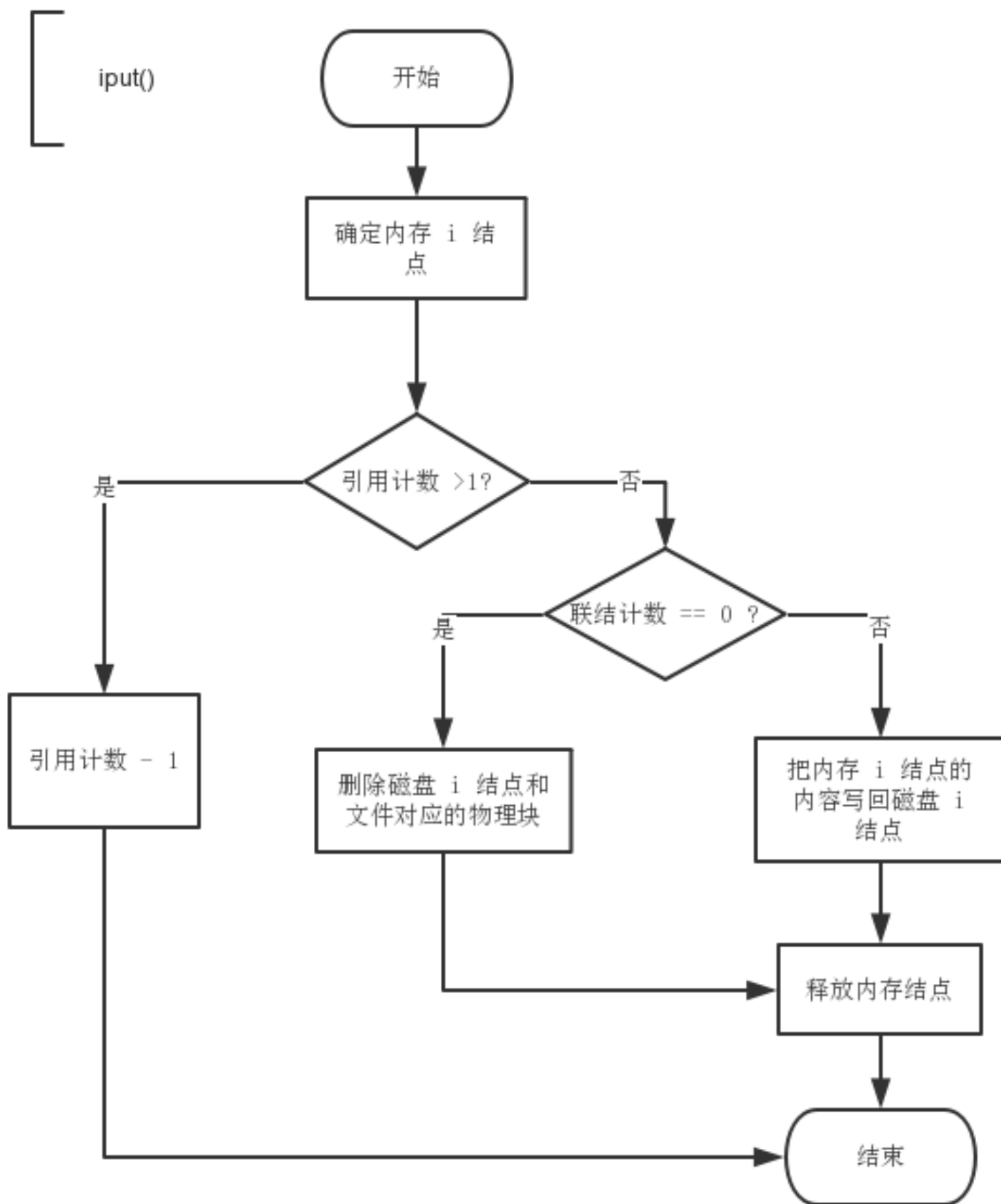
11、 ifree 函数：释放内存节点。



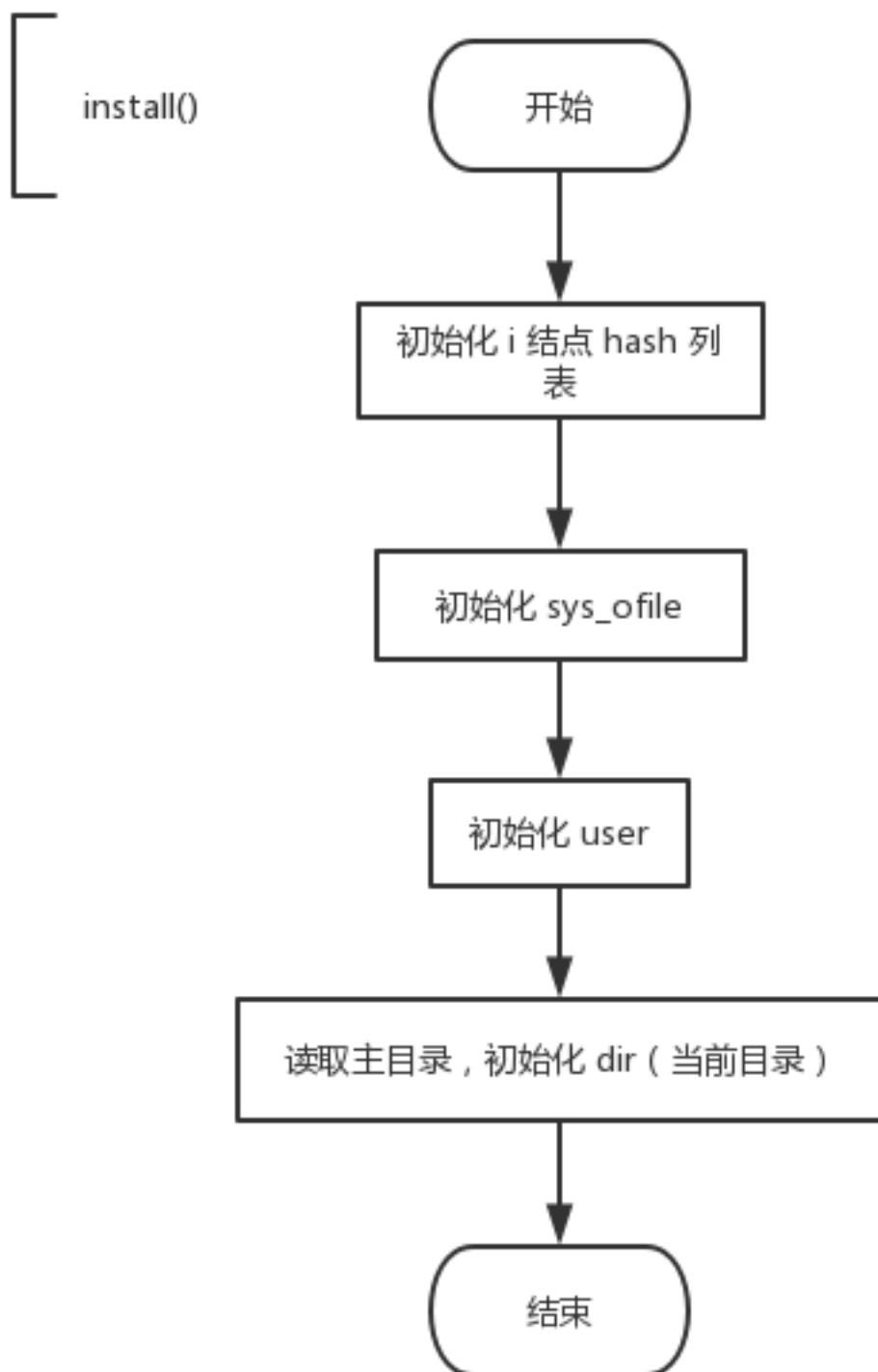
12、 iget()函数：获得内存 i 节点指针。



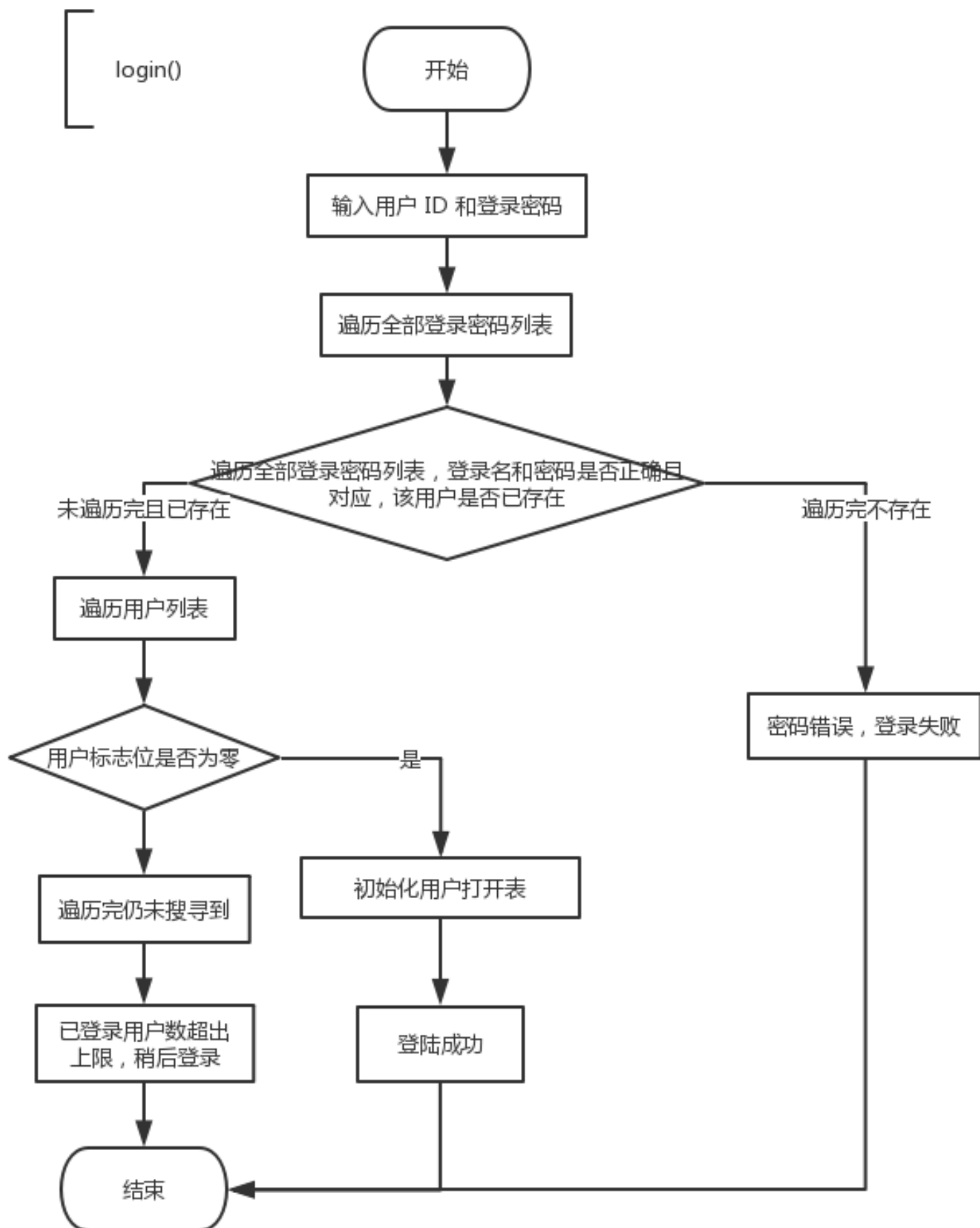
13、 iput()函数：找到内存节点，分情况（引用节点数）释放。



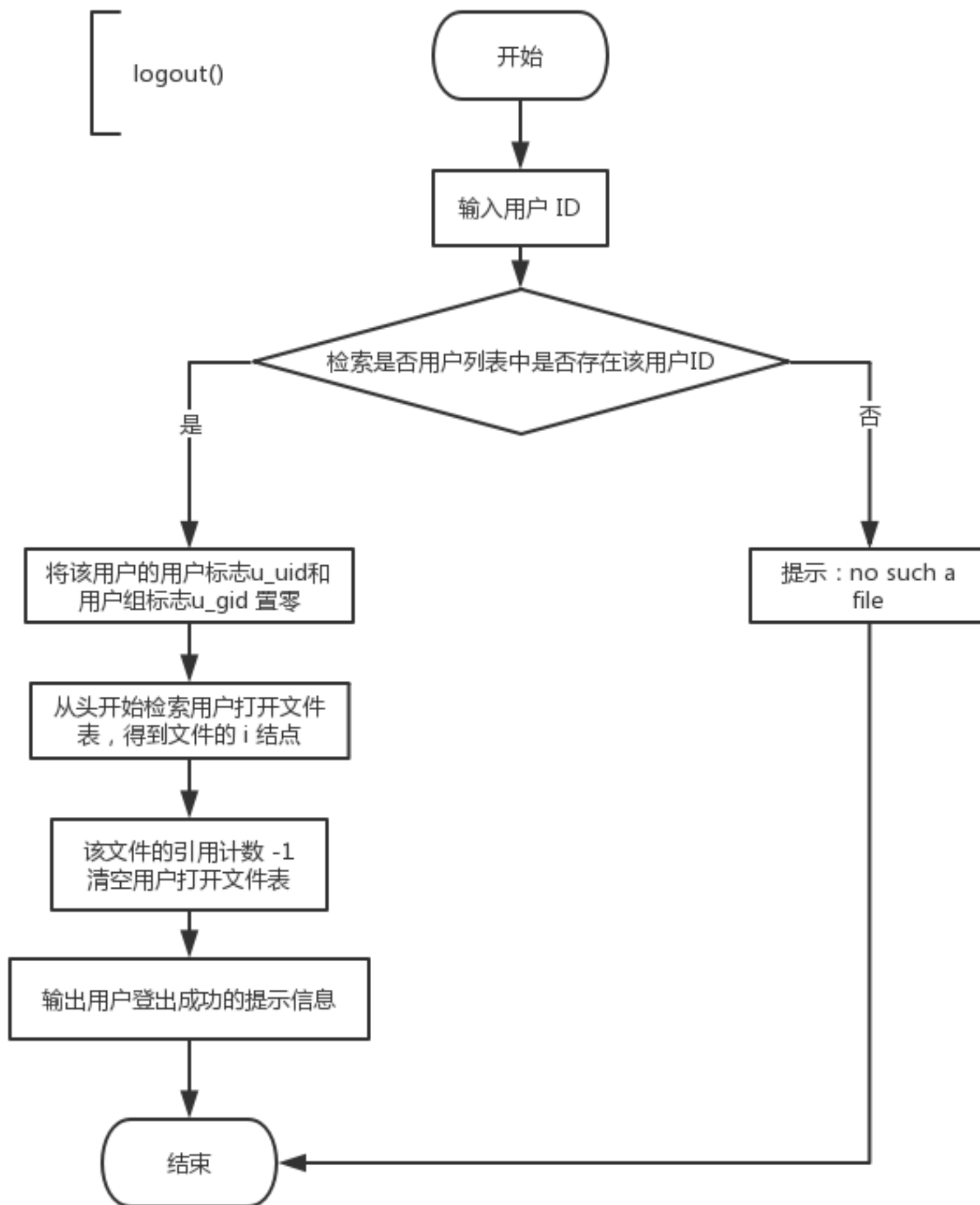
14、 install()函数：初始化内存节点 hash 列表、user、读取主目录，初始化当前目录。



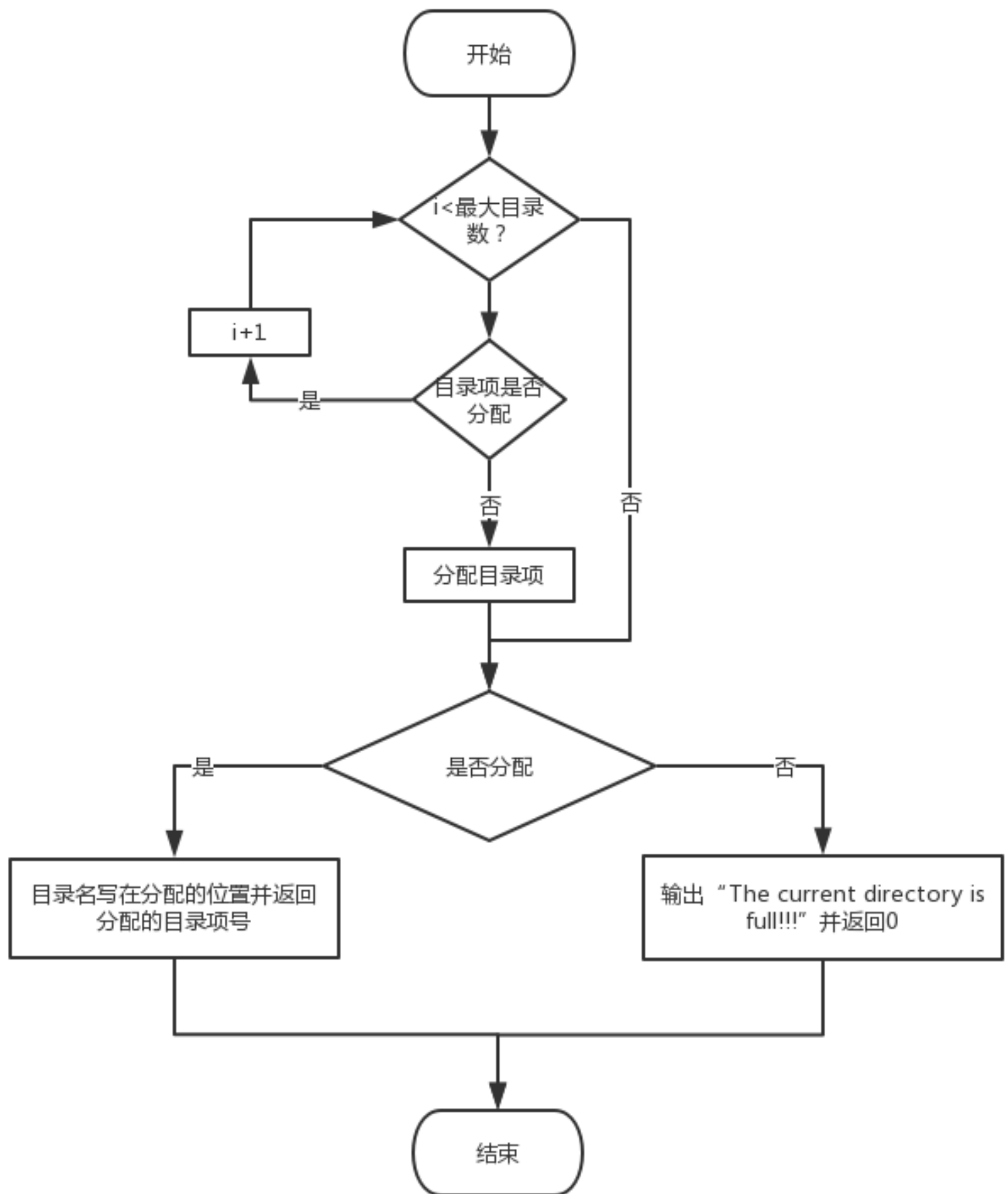
15、 login()函数：登录。



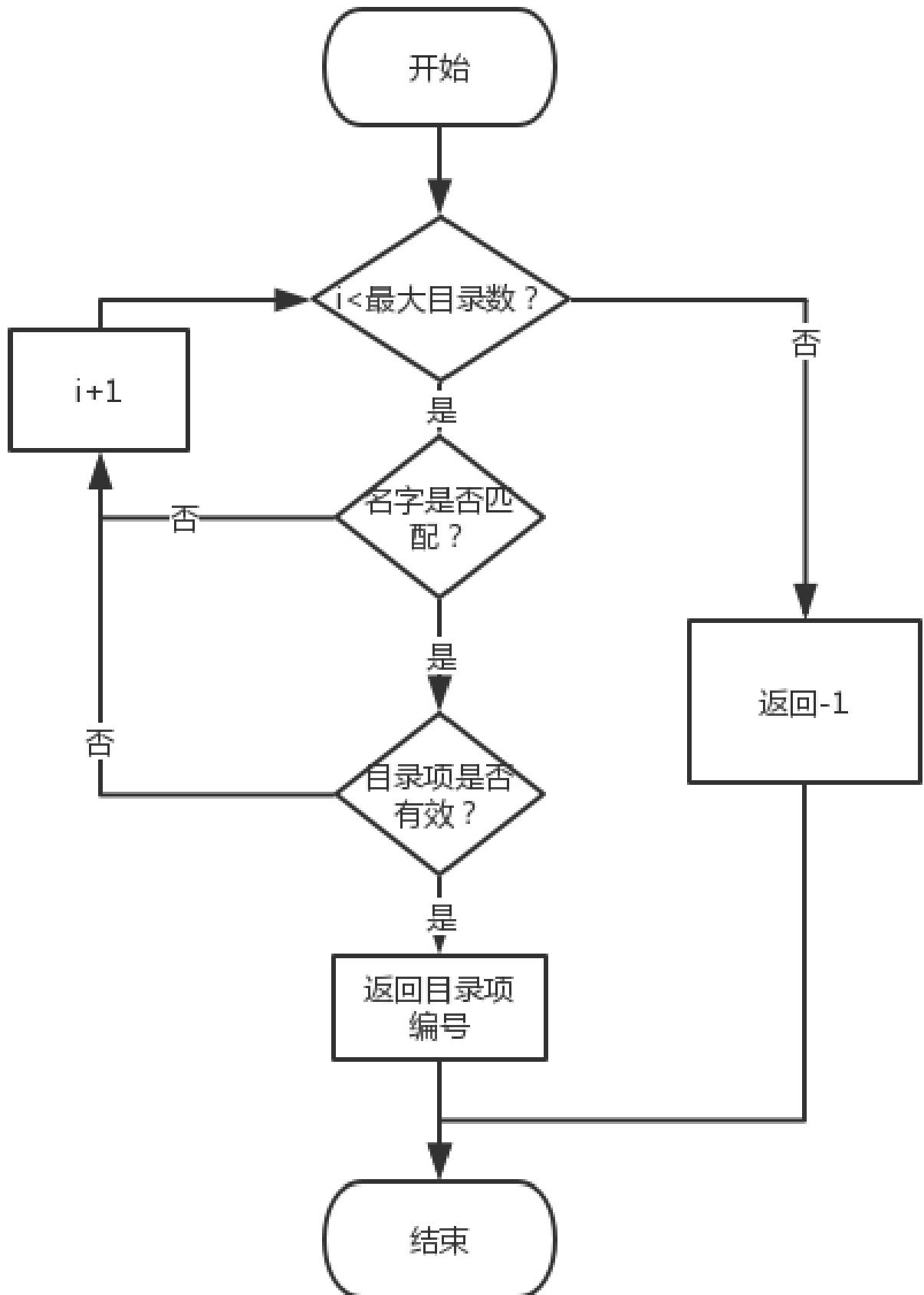
16、 logout()函数：退出登录



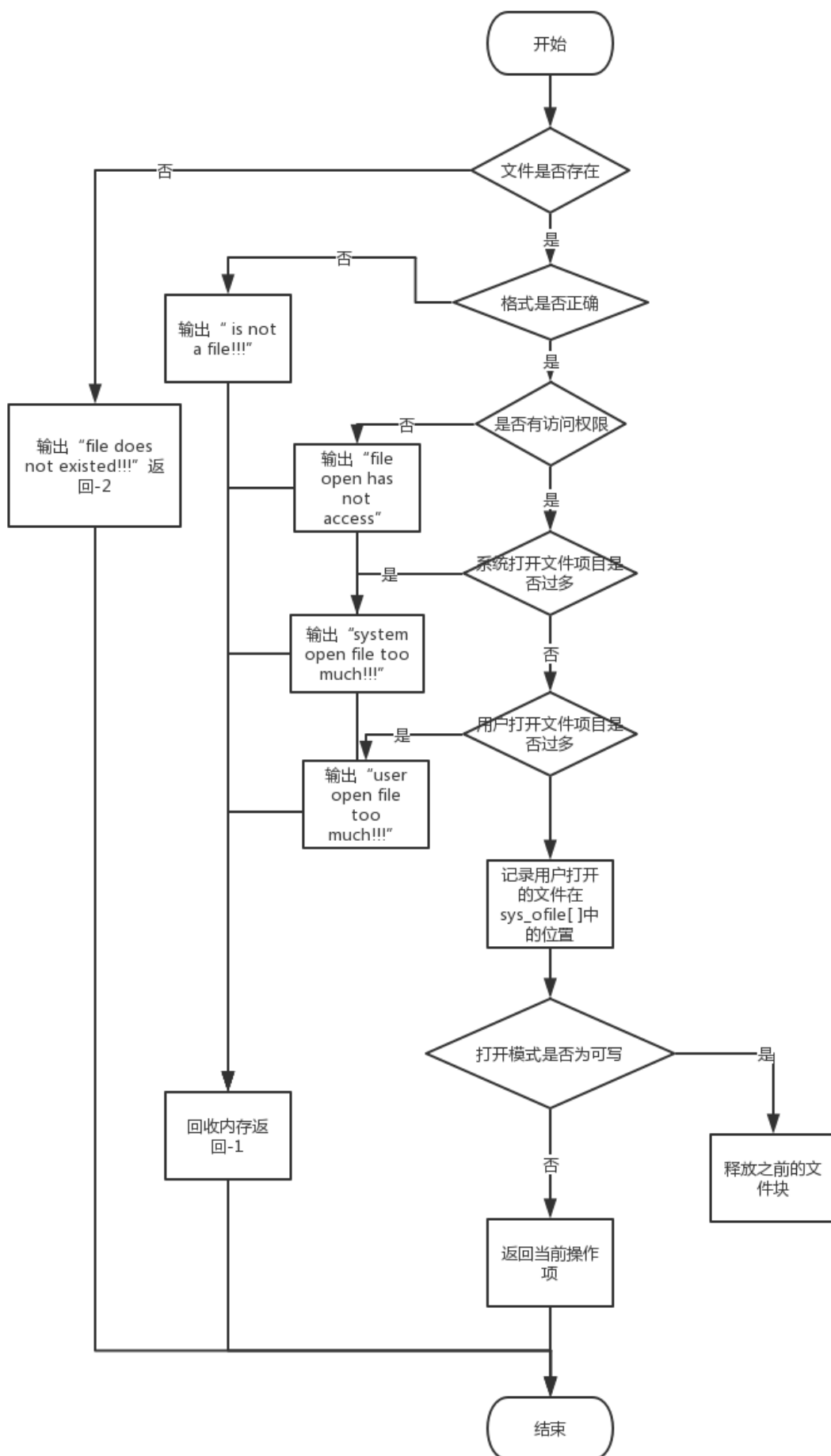
17、 iname 函数：分配目录项。



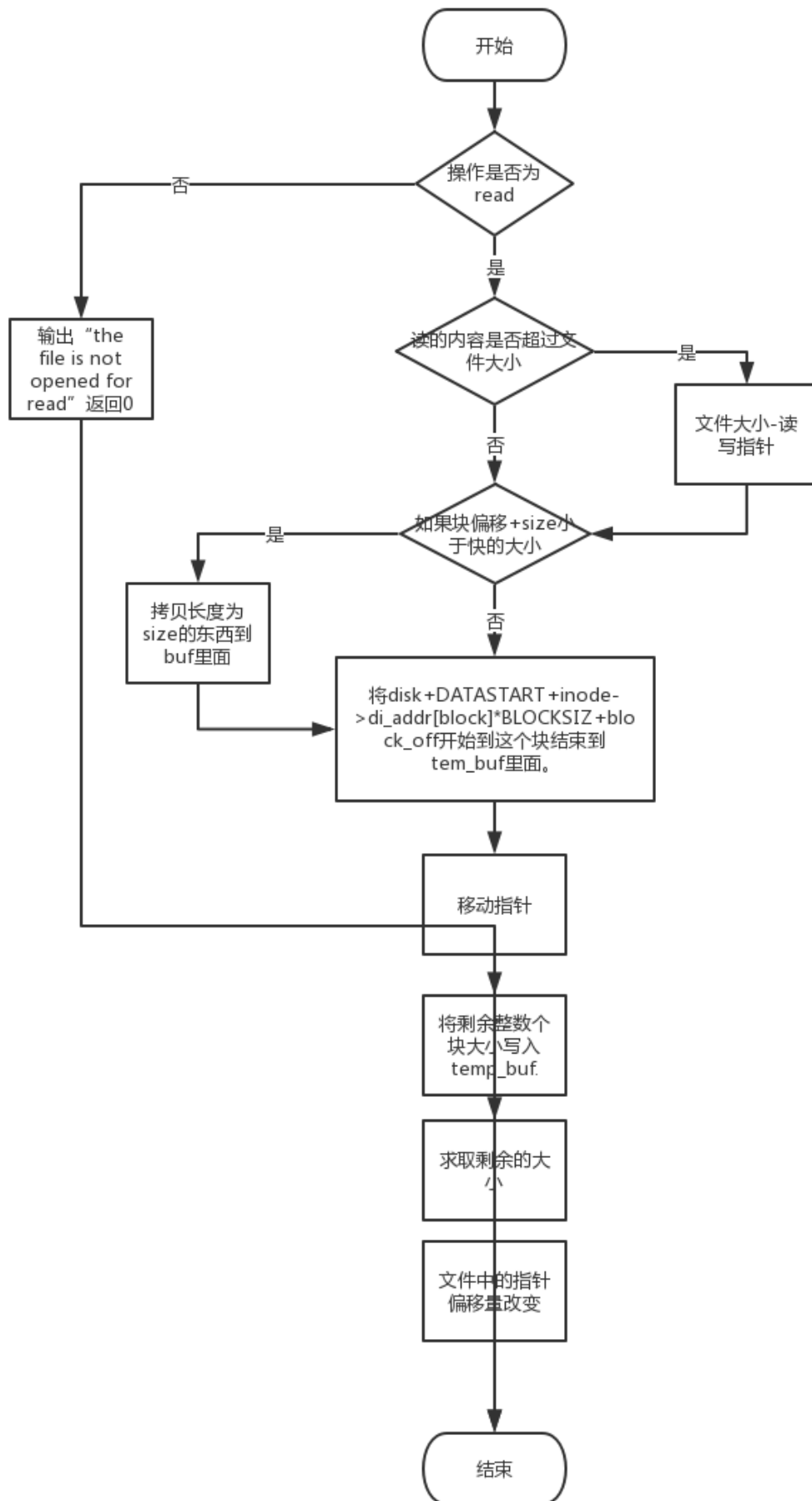
18、 namei 函数：返回目录项编号。



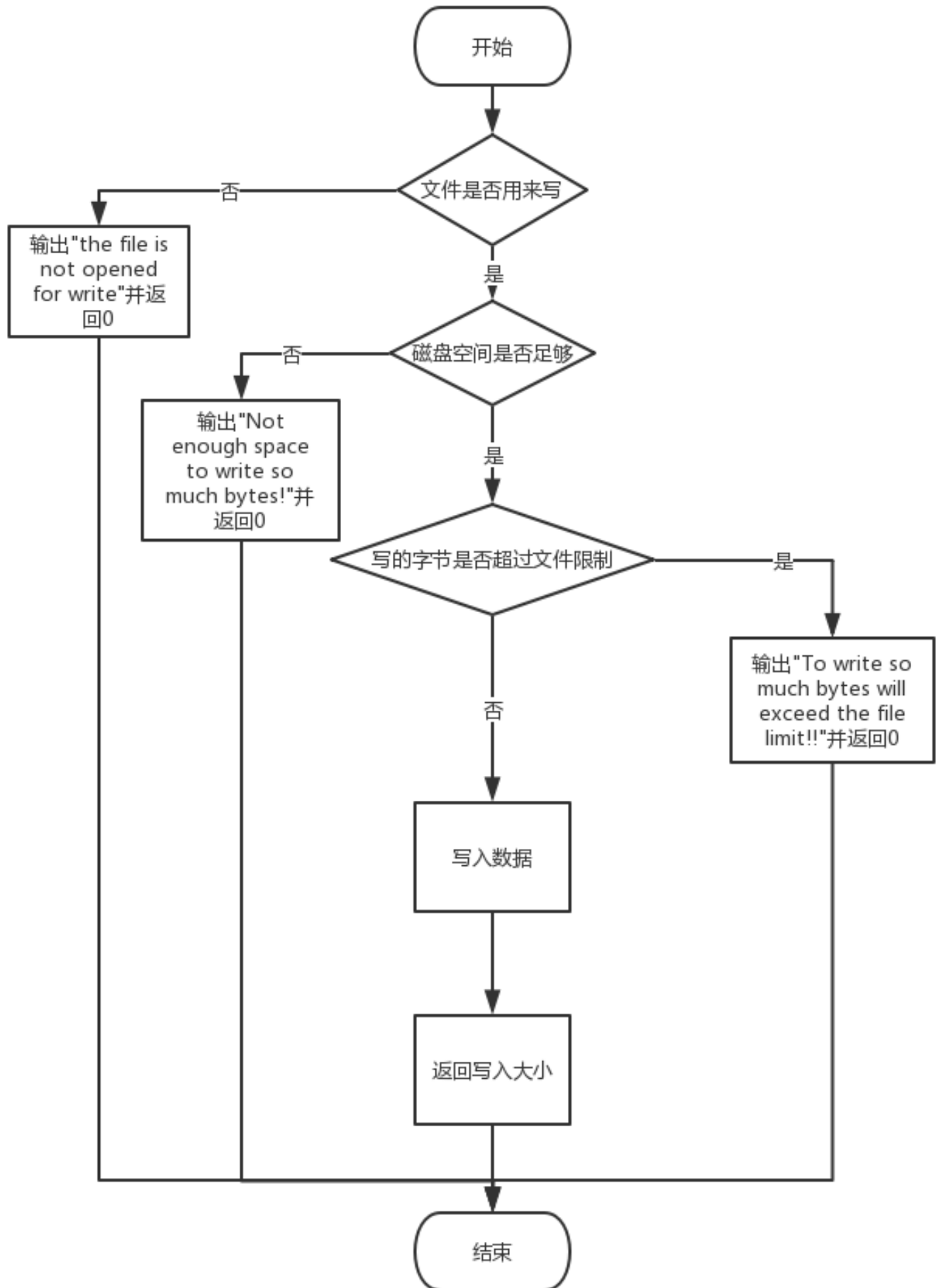
19、 open 函数：打开文件



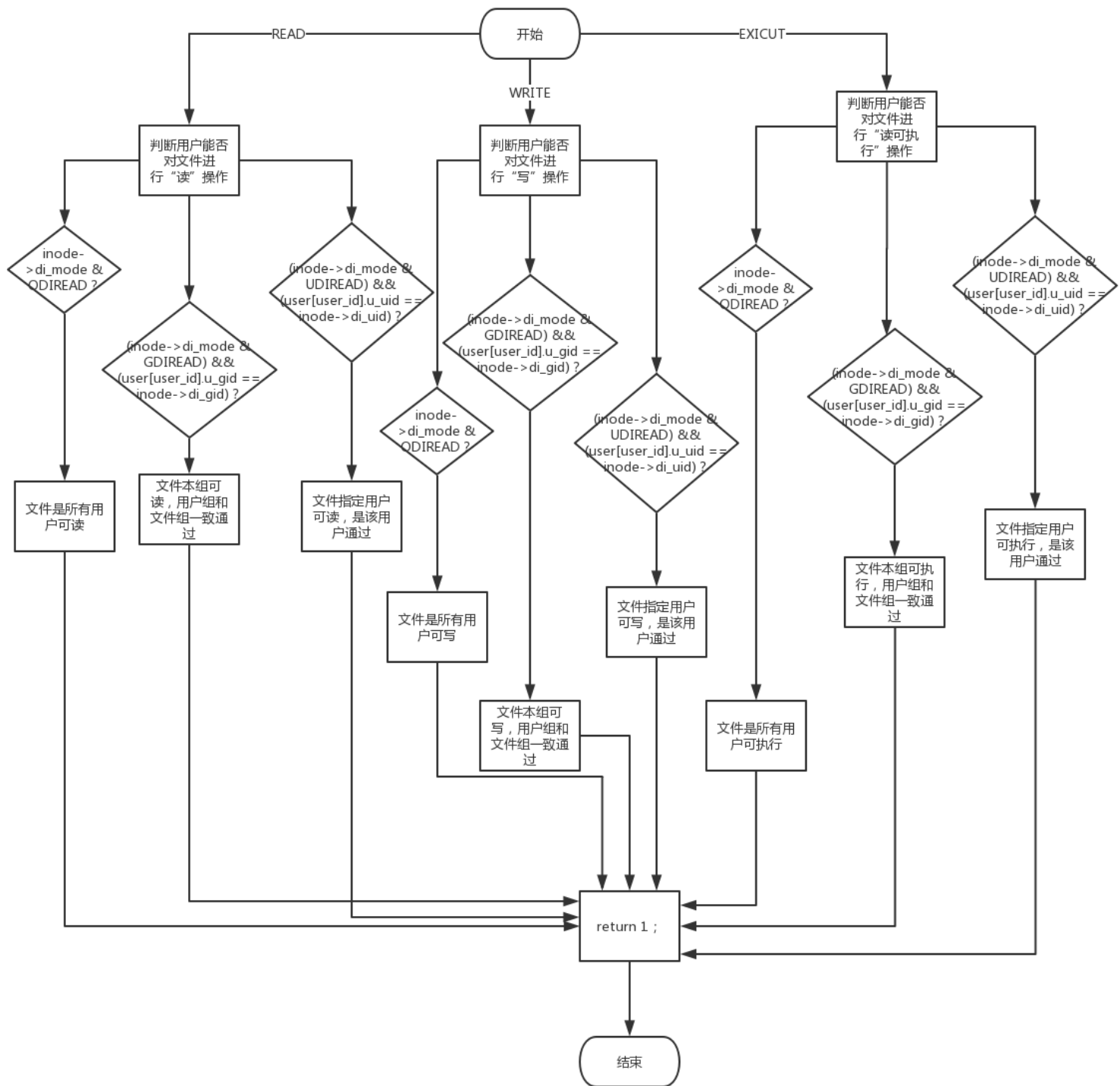
20、 read 函数：执行命令 read。



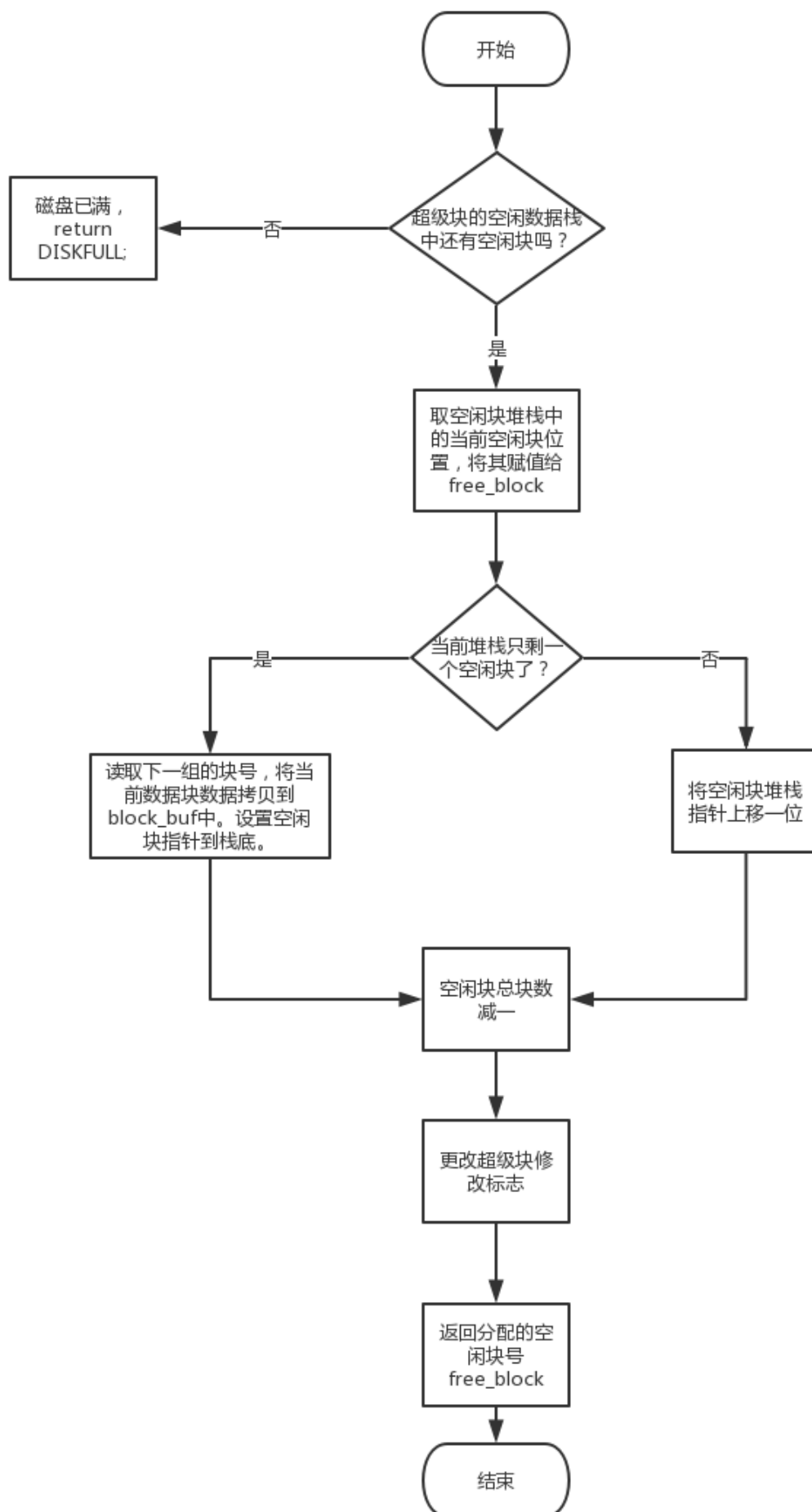
21、 write 函数：执行命令 write。



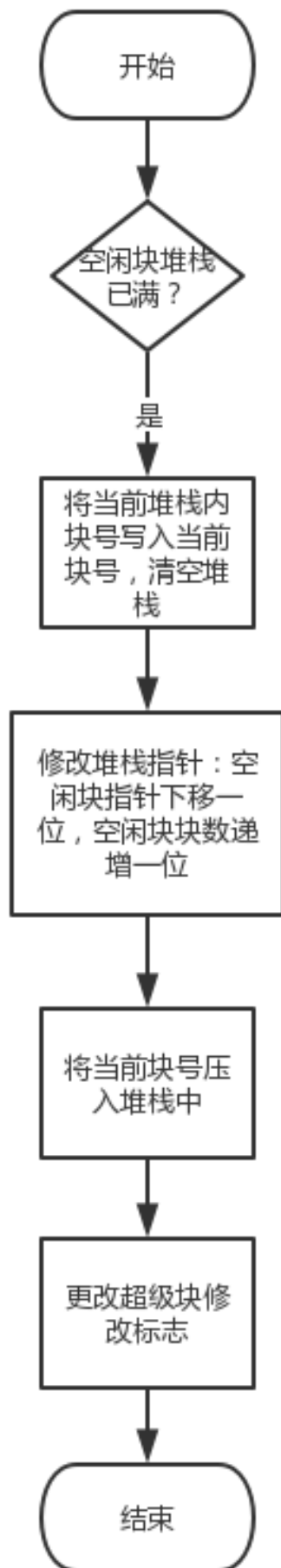
22、 `access` 函数：实现权限判断以及权限分配。



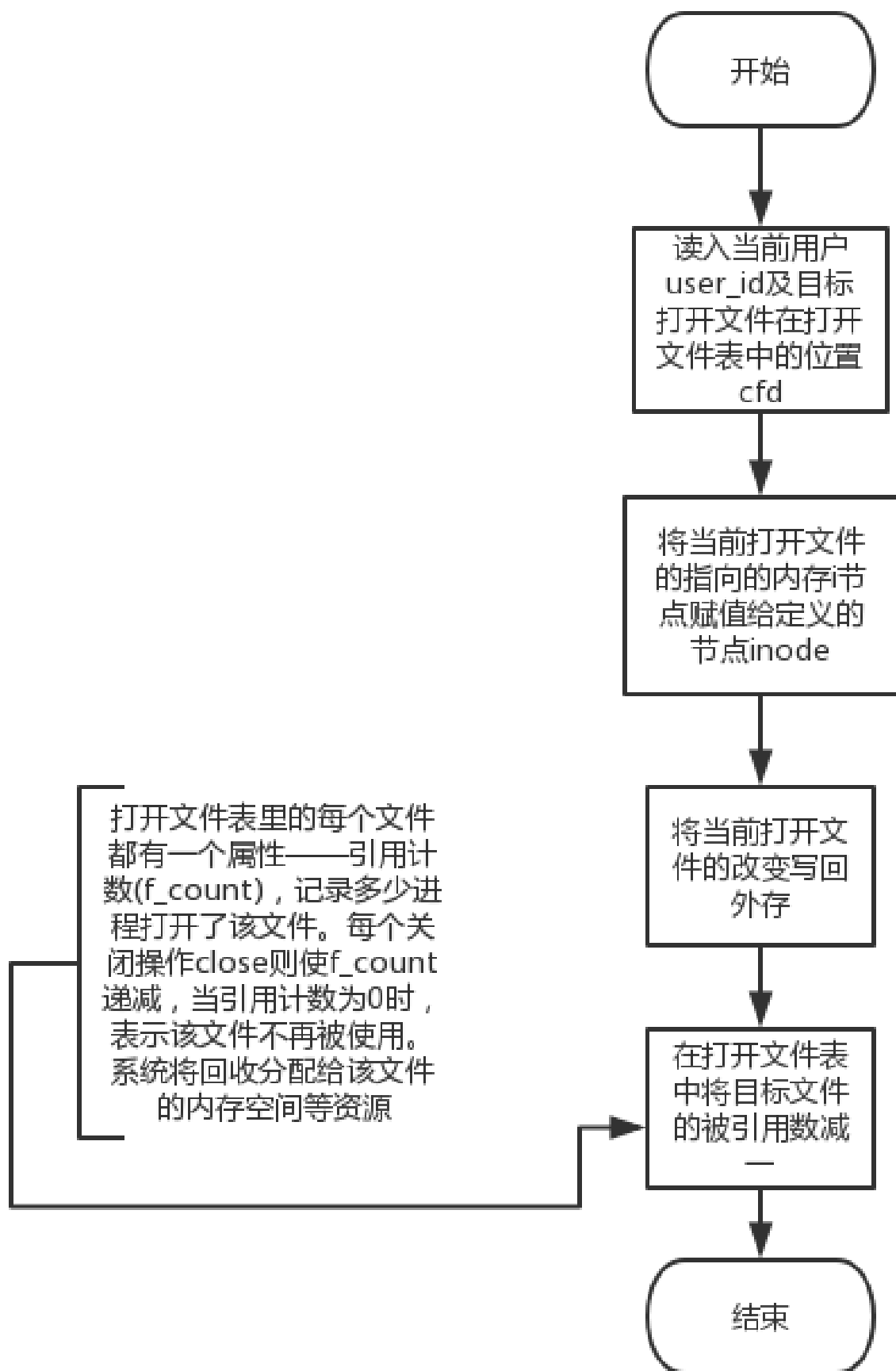
23、 balloc 函数：分配空闲栈中的空闲块。



24、 bfree 函数：在空闲堆栈中释放块。



25、 close 函数：关闭文件。



iii. BUG 调试与成果图。

1、 Write / Read 文件系统中不存在文件时，闪退。

我们在代码中的三处做过改动。地点及原因分别展示如下：

```
short mode,fd;
int cid,size;
token = strtok(str,seps);
if(token == NULL)
    return 1;
cid = getcid(token);
switch(cid){
```

源代码是unsigned short,导致如果打开失败，返回值-1不符合数据类型，从而导致不能运行

```
fd = open(user_id,tstr,char(mode));
if(fd == -1)
{
    printf("new fuction:Wrong input!\n");
    break;
}
```

如果在此处不判断并跳出，则在下面函数传参过程会导致闪退

```
buf = (char*)malloc(size);
size = write(fd,buf,size);
printf("%d bytes have been writed in file %s.\n",size,tstr);
free(buf);
close(user_id,fd);
break;
```

```
fd = open(user_id,tstr,READ);
if(fd == -1)
{
    printf("new fuction:Wrong input!\n");
    break;
}
```

read处出错原因与改正方法均同上

```
buf = (char*)malloc(size+1);
size = read(fd,buf,size);
printf("%d bytes have been read in buf from file %s.\n",size,tstr);
```

2、 逆序输出并且未释放未满盘

仔细分析代码结构，不断输出，寻找问题根源后，我们在代码中的一处做过改动。地点及原因分别展示如下：

```
else
{
    /* 删除磁盘i结点和文件对应的物理块 */
    block_num = pinode->di_size/BLOCKSIZE;
    if(pinode->di_size%BLOCKSIZE)
        block_num++;
    for (i=block_num-1; i>=0; i--) {
        bfree(pinode->di_addr[i]);
    }
    ifree(pinode->i_ino);
}
```

源代码没有这一句，若删除的文件有未装满磁盘，则少删除了一个磁盘

选择倒序删除，则块号小的磁盘最后入栈，在分配时pop ()则是正常顺序

3、 成果图

```
Login:2116
Password:dddd
> write aaa 1000
aaa is not a file!!!
new fuction:Wrong input!
> read aaa 1000
aaa is not a file!!!
new fuction:Wrong input!
> mkfile a
> write a 2000
2000 bytes have been written in file a.
>
> mkfile b
> write b 2000
2000 bytes have been written in file b.
>
> del a
>
> mkfile c
> write c 3000
3000 bytes have been written in file c.
> dir

CURRENT DIRECTORY :..
当前共有5个文件/目录
..          xxxxxxxx i_ino->1      <dir>
.           xxxxxxxx i_ino->1      <dir>
etc         xxxxxxxx i_ino->2      <dir>
c           xxxxxxxx i_ino->4      3000 block chain:  3  4  5  6 11 12
b           xxxxxxxx i_ino->5      2000 block chain:  7  8  9 10
```

BUG1被开心地解决了

BUG2也被开心地解决：delete&&输出

四、 【实验感想】

何茂林：了解了文件管理系统的大体框架和应该实现的功能具体有哪些，在阅读代码时明确了在命令行输入命令时文件管理系统相应的具体操作，在调 bug 后认识到 1.一定要判断 fopen 函数返回的指针值 2.在删除时考虑文件未装满一个盘时的情况，总的来说写实验的时候应当明确每一个操作的具体过程，不要漏掉情况，注意程序的鲁棒性。在小组分工方面，我很开心倩芸给出的有效建议，使每个人都切切实实地参与了实验，学到了东西。我的队友都非常的负责，跟她们合作非常愉快。尤其喜欢有一个人能够和自己有思维共鸣。

王倩芸：我将感想分为三部分：1.与茂林合作画 Main&Shell 文件流程图：一起读懂主干函数，我主图，茂林主流程，一气呵成。2.与茂林合作修复 BUG：学到了从表象寻找函数，利用输出，步步查询问题根源，错因很可能就是一个等号的区别。3.与茂林合作整理队友的流程图，编写实验报告：陪伴是世界上最长情的告白，我爱上了组团编代码的气氛。4.合理的分工合作很重要：在工作的独立性和依赖性之间做平衡是件很重要的事情。

王驰翔：通过此次实验，系统地了解了操作系统文件管理系统中的诸个部分及其功能。通过制作流程图，详细地学习、理解和分析了文件系统中某几个模块的实现方式及其过程。最终收获很多，对文件系统有了更具体的认知。。

段倩薇：通过绘制流程图对 dir 函数可实现的几个功能有了明确的认识，看到了文件，目录的创建过程，对 linux 系统的文件管理部分功能的实现细节有了直观的认识。。

王昊笛：本次实验我负责了 `halt/format/iallfre` 三个 `cpp` 文件的流程图制作。虽然只有三个文件四个函数，但是为了更好的理解程序，还是要对程序的其它部分有一定的了解。本次实验加深了我对文件系统的理解，是对理论课很好的补充。在学习之余更让我感受到小组学习的乐趣，收获颇丰。

袁玥：文件系统包含很多方面的知识，数据项记录以及文件的结构组成，外存的分配方式，目录的管理还有文件存储空间的管理。根据老师给的代码去学习了解系统的各个功能函数具体是如何实现的，画出他们的流程图让我对各个函数以及整体系统更加了解了，也注意到了之前没有注意到的小细节。例如为什么要专门用一个打开文件表来保存打开文件位置而不是直接将文件位置放在索引节点中。这次的实验让我梳理了文件管理方面的知识点，以后还需更多的学习才能进一步了解操作系统的原理。

戚镇麒：通过在模拟实验对几个函数的流程绘制，对 Linux 文件系统的 `open`, `read`, `write` 和检索函数的实现有了更加深刻的认识。