// 以Create为例

int CREATE(string name)

{

int index;

if (path == "")

{

cout << "请在文件夹中创建文件" << endl;

return 0;

}

for (index = 0; index < cur\_user.next->cur\_user\_direct\_size; index++)

{

if (path == cur\_user.next->direct[index].directname)

{

break;

}

}

int i;

for (i = 0; i < cur\_user.next->direct[index].cur\_file\_size; i++)

{

if (name == cur\_user.next->direct[index].ufd[i].filename)

{

break;

}

}

if (i < cur\_user.next->direct[index].cur\_file\_size)

{

cout << "文件名重复" << endl;

return 0;

}

if (cur\_user.next->direct[index].cur\_file\_size == 16)

{

cout << "该文件夹文件已达到16个" << endl;

return 0;

}

int j;

for (j = 0; j < 64; j++) //判断是否有空闲块

{

if (fat[j].used == 0)

break;

}

if (j >= 64)

{

cout << "磁盘没有空闲块了" << endl;

return 0;

}

cur\_user.next->direct[index].ufd[cur\_user.next->direct[index].cur\_file\_size].filename = name;

cur\_user.next->direct[index].ufd[cur\_user.next->direct[index].cur\_file\_size].addr = j; //文件起始盘块号

cur\_user.next->direct[index].ufd[cur\_user.next->direct[index].cur\_file\_size].length = 0;

cur\_user.next->direct[index].ufd[cur\_user.next->direct[index].cur\_file\_size].protect\_code = 2; //表示可读可写

cur\_user.next->direct[index].cur\_file\_size++; //用户文件数量加1

return 1;

}