**项目描述**：

**1**输入/输出系统

物理磁盘是一个由柱面、柱面里的磁道、磁道里的扇区和扇区里的字节所组成的二维结构。输入/输出系统的任务是，通过将磁盘表示为一个标号从0到L-1的逻辑块的线性系列来隐藏其二维组织方式，其中L是在物理磁盘上磁盘块的总数目。

我们将用一个数组ldisk[L][B]来模拟磁盘，其中L是逻辑块数，B是块长度，即每个块的字节数。输入/输出系统的任务就是从文件系统接收逻辑块号，并将相应的块读入或者写入由命令指定的内存区域中。

使用如下两个函数来定义文件系统和输入/输出系统之间的接口，只要系统读写磁盘块，它就要调用这两个函数：

1. read\_block(int i, char \*p)：拷贝逻辑块ldisk[i]到以指针p指定的位置开始的内存中。拷贝的字符数和块长度B相等。
2. write\_block(int i, char \*p)：从以指针p指定的位置所开始的内存中，拷贝和块长度B相等的字符数到逻辑块ldisk[i]中。

**2**用户和文件系统之间的接口

文件系统必须支持以下函数：create、destory、open、read、write、lseek和directory。

1. create(symbolic\_file\_name)： 使用指定的名称创建一个新文件。
2. destory(symbolic\_file\_name): 删除指定名称的文件。
3. open(symbolic\_file\_name): 为了读和写而打开指定名称的文件；返回一个索引值，供随后的read、write、lseek或者close操作使用。
4. close(index): 关闭指定文件。
5. read(index, mem\_area, count): 从指定文件中顺序读取一定数目的字节到内存中。被读的字节数在count中指定，开始的内存地址在mem\_area中。读操作从文件的当前位置开始。
6. write(index, mem\_area, count)： 把从mem\_area开始的内存中的一定数目的字节写入到指定的文件中。如同使用读操作一样，字节数在count中指定，写操作从文件当前位置开始。
7. lseek(index, pos)： 将文件的当前位置移动到pos，其中pos是从文件开始的指定的字节数的一个整数。当文件最初被打开时，当前位置自动设为0。每一次读或写操作后，它马上指向跟在最后被访问字节的后一个字节。
8. directory： 列出所有文件的名字及其长度。

3文件系统的组织

磁盘的前k个磁盘块是保留的；它们包括以下描述信息：



Bitmap说明了哪些磁盘块是空闲的以及哪些已被现有文件占用了。位图中的每一位对应于一个逻辑磁盘块。只要写操作使得文件增加或者当删除文件时，文件系统都要查询位图。

前k个磁盘块中剩余的部分包括了一个**文件描述符**数组。描述符的最大数由块的长度和k的数目来决定。每一个描述符都包括以下信息：

* 以字节表示的文件长度
* 含有文件内容的磁盘块的块号的一个数组。

**4**目录

用一个目录记录所有的文件。它对应于磁盘上的第一个文件描述符（见图）。初始没有文件时，目录的长度为0，没有分配磁盘块。当创建文件时，目录就会扩充。目录采用一个记录项的数组来组织文件。每个记录项包括文件名和文件描述符索引两部分。

**实验内容：**设计并实现I/O和文件系统。

**实验要求：**

1.设计并实现模拟的输入/输出系统，特别是两个函数 read\_block(int i, char \*p) 和 write\_block(int i, char \*p)。

2.设计并实现在输入/输出系统上层的文件系统。它应该支持定义用户/文件系统接口的 8个函数。

3.为了说明文件系统行为的各个方面，可以自行编写测试命令来测试文件系统。

4..用C/C++实现，提交源代码、实验报告（markdown格式）和视频报告，在关键代码处给出注释。实验报告应至少包含设计思路、测试结果等内容。

5.视频报告：提交人讲解实验设计、代码及测试实验演示，时间控制在5分钟以内。

6.评分标准：代码及结果（5分）、实验报告（2分）、视频讲解（3分）。