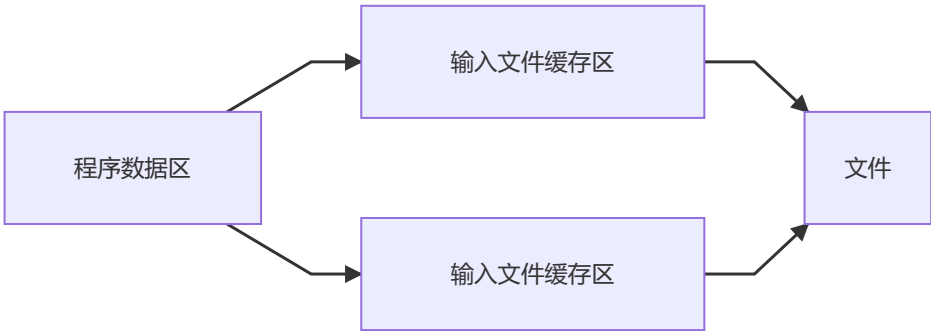


基础作业

1 FILE类型结构体实际上是指文件缓冲区，这个结构体当中包含了缓冲区基址指针、当前读写位置指针、缓冲区容量和文件实际大小等信息。画出文件缓冲区的示意图

- 缓冲区大小为4096
-



2 如何打开和关闭一个文件？如何定位打开文件时触发了什么错误？

- 打开文件：

```
1 | fopen("文件路径", "文件方式")
```

- 关闭文件

```
1 | fclose(FILE指针名)
```

- 在每个函数操作下插入该代码，如果出现error则说明出错

```
1 | if(NULL==fp)
2 | { //如果执行失败则会返回错误码，
3 |   //perror的作用是返回上面函数的错误码。
4 |   perror("fopen");
5 |   goto error;
6 | }
```

3 r、r+、rb+模式有什么区别？从读、写和偏移的角度来讨论

打开模式	只读	只可以写	读写兼备
文本模式	r		r+
二进制模式			rb+

- 对于r+而言无论是读取内容还是写入内容，都可在文件中任意位置进行，且进行写入操作时，会覆盖原有位置的内容。

4 什么时候缓冲区的内容会写回磁盘？

- 缓冲区被写满满
- 执行fclose的时候
- fflush(fp) 主动刷新

5 路径是什么？有哪几种路径？

- 路径是什么
 - 磁盘上寻找文件时，所历经的文件夹线路叫路径
- 有哪几种路径
 - 相对路径和绝对路径
 - 另外
 - 相对路径就是相对于当前文件的路径。网页中一般表示路径使用这个方法。
 - 绝对路径就是你的文件或目录在硬盘上真正的路径。

6 整理所有已经学习过的读写接口

函数名	读取	写入	备注
fgetc()	✓		每次读取一个字符
fputc()		✓	每次写入一个字符
fwrite()		✓	需要输出数据的地址
fread()	✓		需要读取数据的地址

7 如何偏移文件的读写位置

- rewind()将文件指针重新指向一个流的开头。
 - 如果一个文件具有读写属性，则我们写完文件时，文件指针指向结尾，当需要再次读取时就需要用到rewind函数。
- fseek() 偏移文件指针 成功返回 0，失败返回-1
 - 其中使用
 - SEEK_SET:表示文件开头
 - SEEK_CUR:表示文件当前位置
 - SEEK_END:表示文件末尾