

# IO总结

## open函数

```
/*
原型: int open(const char *pathname, int flags, mode_t mode)
使用: 路径/文件, 权限, 模式
说明: 返回一个打开文件的文件标识符并返回, 打开失败则返回-1
*/
int fd;
fd = open("a.c", O_CREAT | O_RDWR, 0666);
//给予可执行+可读可写的权限
```

## close函数

```
/*
原型: int close(int fd);
使用: 关闭文件标识符下的文件, 无返回值
*/
close(fd);
```

## read函数

```
/*
原型: ssize_t read(int fd, void *buf, size_t count);
使用: 文件标识符, 读取存贮的字符数组buf[], 32位。读取错误时返回-1。
说明: 使用时光标指针会随着read变。
*/
int fd;
char buf[32]={0};
ssize_t ret;

fd = open("a.c", O_RDWR);
ret = read(fd, buf, 32);
//ret为字节数, buf为内容
```

## write函数

```

/*
原型: ssize_t write(int fd, const void *buf, size_t count);
使用: 文件标识符, 内容, 内容的字节数
说明: 通过open+read+write可以实现复制功能, 写入失败返回-1。
*/
int fd;
char buf[32]={0};
ssize_t ret;

fd = open("a.c",O_RDWR);
write(fd,"hello",5);
//ret为字节数, buf为内容

```

## lseek函数

```

/*
原型: off_t lseek(int fd, off_t offset, int whence);
使用: 文件标识符, 偏移量, 位置
offset: 可正可负, 正值向前移动, 负值向后移动
where: SEEK_SET:相对于文件开头
        SEEK_CUR:相对于当前读写指针位置
        SEEK_END:相对于文件末尾
说明: lseek成功返回偏移量, 失败返回-1。
*/
int fd;
fd = open("a.c",O_RDWR);
lseek(fd,0,SEEK_SET);
//ret为字节数, buf为内容

```

## opendir函数

```

/*
原型: DIR *opendir(const char *name);
        opendir前带个*, 表示为指针, 有返回指针参数以便其它IO函数使用
使用: 通过终端输入argv, 路径名称等
说明: On error, NULL is returned
*/
DIR *dp;
dp=opendir(argv[1]);

```

## closedir函数

```
/*
原型: int closedir(DIR *dirp);
*/
DIR *dp;
closedir(dp);
```

## mkdir函数

```
/*
原型: mkdir [OPTION]... DIRECTORY...
使用: 通过终端输入argv, 确定目录名称
*/
mkdir(argv[1], 0666);
```

## readdir函数

```
/*
原型: struct dirent *readdir(DIR *dirp);
说明: 返回值不为空即使用成功
*/
int ret;
DIR *dp;
struct dirent *dir;
dir = readdir(dp);
```

## stat函数

```
/*
原型: int stat(const char *path, struct stat *buf);
使用: 获取文件信息
说明: 成功返回0, 失败返回-1
*/
struct stat filestat;
int message;
message = stat(argv[1], &filestat);
printf( "the size of %s is %ld\n", argv[1], filestat.st_size );
```

## getuid函数

```
原型: uid_t getuid(void);
说明: int型, 返回真实使用者的id。
```

# getpwuid函数

```
/*  
原型: struct passwd *getpwuid(uid_t uid);  
使用: 通过用户的uid查找用户的passwd数据  
说明:  
*/  
    struct passwd *my_info;  
    my_info = getpwuid( getuid() );  
    printf( "my name = [%s]\n", my_info->pw_name );
```