

文件类型

- 普通文件：存放在磁盘里面的文件
- 目录文件：存放在磁盘里面，其他文件的位置信息
- 链接：存放另一个文件的路径
- 设备文件：字符设备 块设备
- 管道
- socket

磁盘

万物皆文件

鼠标

文件指针

ISO C 标准.

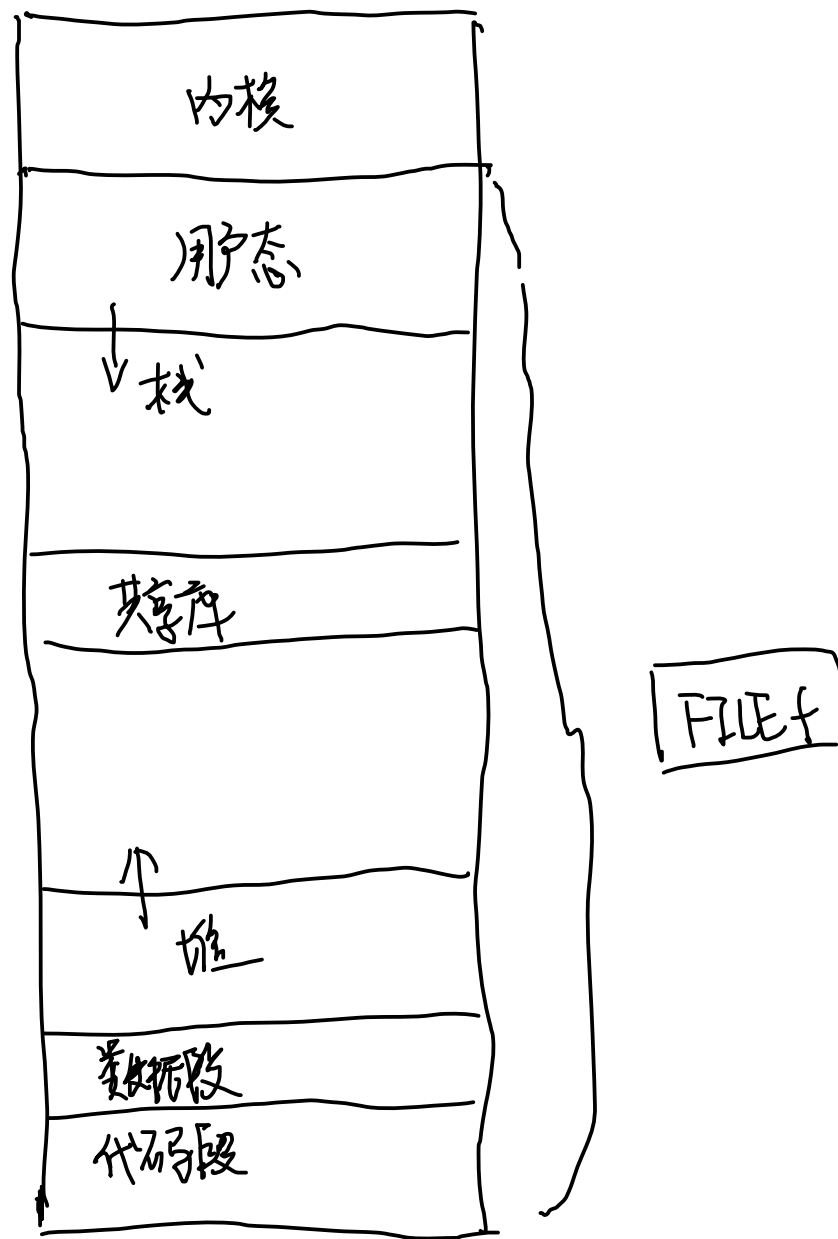
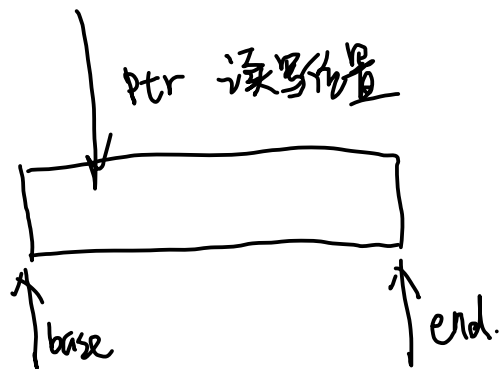
可移植性

FILE *

如何查看 FILE

1. 预处理

2. grep



打开关闭文件

fopen FILE *fopen(const char *pathname, const char *mode);

fclose

mode 的可选模式列表

模式	读	写	位置	截断原内容	创建
rb	Y	N	文件头	N	N
rb+	Y	Y	文件头	N	N
wb	N	Y	文件头	Y	Y
wb+	Y	Y	文件头	Y	Y
ab	N	Y	文件尾	N	Y
ab+	Y	Y	文件尾	N	Y

Linux内,文本模式与二进制模式 没有区别的

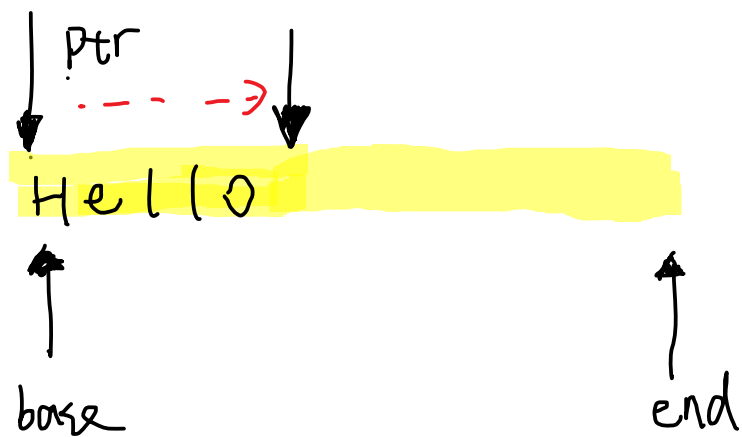


func.h

```
#include <stdio.h>
#include <string.h>
#define ARGS_CHECK(argc,val) {if(argc != val) {printf("args error!\n"); return -1;}}
#define ERROR_CHECK(ret,val,info) {if(ret == val) {perror(info);return -1;}}
```

追加模式

"a" "a+"



第一次写入时, 自动偏移到文件末尾

```
1 #include <func.h>
2
3 int main(int argc, char *argv[])
4 {
5     ARGS_CHECK(argc, 2);
6     FILE *fp;
7     fp = fopen(argv[1], "ab+");
8     ERROR_CHECK(fp, NULL, "fopen");
9     char buf[128] = {0};
10    fread(buf, sizeof(char), sizeof(buf), fp);
11    puts(buf);
12    const char *p = "how are you";
13    fwrite(p, sizeof(char), strlen(p), fp);
14    fclose(fp);
15    return 0;
16 }
17
```

```
long ftell(FILE *stream);
```

```
#include <func.h>

int main(int argc, char *argv[])
{
    ARGS_CHECK(argc, 2);
    FILE *fp;
    fp = fopen(argv[1], "ab+");
    ERROR_CHECK(fp, NULL, "fopen");
    long ret;
    char buf[128] = {0};
    ret = ftell(fp);
    printf("ptr pos = %ld\n", ret);
    fread(buf, sizeof(char), sizeof(buf), fp);
    puts(buf);
    ret = ftell(fp);
    printf("ptr pos = %ld\n", ret);
    const char *p = "how are you";
    fwrite(p, sizeof(char), strlen(p), fp);
    ret = ftell(fp);
    printf("ptr pos = %ld\n", ret);
    fclose(fp);
    return 0;
}
```