

# 2024 操作系统实验（二）

本次实验重点在于：熟悉掌握FAT12文件系统、gcc+nasm联合编译，了解实模式与保护模式的基本内容。

## 实验内容 - FAT12镜像查看工具

用C/C++和nasm编写一个FAT12镜像查看工具，读取一个.img格式的文件并响应用户输入。

### 功能列表

1. 运行程序后，读取FAT12镜像文件，并提示用户输入指令
2. 用户输入 `ls 路径`，输出根目录及其子目录的文件和目录列表。
  1. 对每个目录路径，首先输出路径名，加一个冒号:，换行，再输出路径下的文件和目录列表。路径下的目录包括 `.` 和 `..`（参考Linux系统中的 `ls -a` 命令）
  2. 使用红色(033[31m)颜色输出目录名，不添加特殊颜色输出文件的文件名。
  3. 当用户不添加任何选项执行ls命令时，每个文件/目录项之间用两个空格隔开。
  4. 当用户添加 `-l` 参数时，
    1. 在路径名后，冒号前，另输出此目录下直接子目录和直接子文件的数目，两个数字之间用空格连接。两个数字不添加特殊颜色
    2. 每个文件/目录占据一行，在输出文件/目录名后，空一格，之后：
      1. 若项为目录，输出此目录下直接子目录和直接子文件的数目，两个数字之间用空格连接。这两个数字不添加特殊颜色
      1. 不输出 `.` 和 `..` 目录的子目录和子文件数目
      2. 若项为文件，输出文件的大小（以字节为单位）
  3. 对于 `-l` 参数用户可以在命令的任何位置、任意多次地设置-l参数，但只能设置一次路径名
  4. 直接子目录不计算 `.` 和 `..`
  5. 当用户给出不支持的命令参数时，报错，给出错误提示
  6. 当用户不设定路径时，默认路径为镜像文件根目录
  7. 路径应当支持任意多级的相对路径，注意：

1. 在根目录后任意多级 `..` 仍表示根目录（即，**根目录的父目录即为根目录**），如 `/HOUSE/ROOM/../../../../..` 路径即为根目录 `/`
2. 在解析路径时**不能视为简单的字符串预处理**，如，`/HOUSE/A/..` 其实就等价于 `/HOUSE`，但是当 `/HOUSE` 路径下不存在 `A` 目录时，该命令应该报错（参考Linux系统中的 `ls` 命令）
3. 用户输入 `cat` 文件名，输出文件名**对应文件的内容**，若路径**不存在或不是一个普通文件则报错，给出提示**，提示内容不做严格限定，但必须体现出错误所在。
4. 用户输入 `exit`，退出程序。

## 示例

假设镜像文件的结构如下图所示：



输入 `ls`：

```
> ls
/:
HOUSE NJU ROLL.TXT
/HOUSE/:
. . . ROOM
/HOUSE/ROOM/:
. . .
/NJU/:
. . . ABOUT.TXT CS SOFTWARE
/NJU/CS/:
. . .
/NJU/SOFTWARE/:
. . . SE1.TXT SE2.TXT
```

输入 `ls -l`：

```
> ls -l
/ 2 1:
HOUSE 1 0
NJU 2 1
ROLL.TXT 1945

/HOUSE/ 1 0:
.
..
ROOM 0 0

/HOUSE/ROOM/ 0 0:
.
..

/NJU/ 2 1:
.
..
ABOUT.TXT 1390
CS 0 0
SOFTWARE 0 2

/NJU/CS/ 0 0:
.
..

/NJU/SOFTWARE/ 0 2:
.
..
SE1.TXT 48
SE2.TXT 49
```

输入 `ls /NJU -l`:

```
> ls /NJU -l
/NJU/ 2 1:
.
..
ABOUT.TXT 1390
CS 0 0
SOFTWARE 0 2

/NJU/CS/ 0 0:
.
..

/NJU/SOFTWARE/ 0 2:
.
..
SE1.TXT 48
SE2.TXT 49
```

输入 `cat ROLL.TXT`:

> cat ROLL.TXT

Rick Astley - Never Gonna Give You Up

We're no strangers to love  
You know the rules and so do I  
A full commitment's what I'm thinking of  
You wouldn't get this from any other guy  
I just wanna tell you how I'm feeling  
Gotta make you understand  
Never gonna give you up never gonna let you down  
Never gonna run around and desert you  
Never gonna make you cry never gonna say goodbye  
Never gonna tell a lie and hurt you  
We've known each other for so long  
Your heart's been aching but you're too shy to say it  
Inside we both know what's been going on  
We know the game and we're gonna play it  
And if you ask me how I'm feeling  
Don't tell me you're too blind to see  
Never gonna give you up never gonna let you down  
Never gonna run around and desert you  
Never gonna make you cry never gonna say goodbye  
Never gonna tell a lie and hurt you  
Never gonna give you up never gonna let you down  
Never gonna run around and desert you  
Never gonna make you cry never gonna say goodbye  
Never gonna tell a lie and hurt you  
(Ooh give you up)  
(Ooh give you up)  
(Ooh) never gonna give never gonna give (give you up)  
(Ooh) never gonna give never gonna give (give you up)  
We've known each other for so long  
Your heart's been aching but you're too shy to say it  
Inside we both know what's been going on  
We know the game and we're gonna play it  
I just wanna tell you how I'm feeling  
Gotta make you understand  
Never gonna give you up never gonna let you down  
Never gonna run around and desert you  
Never gonna make you cry never gonna say goodbye  
Never gonna tell a lie and hurt you  
Never gonna give you up never gonna let you down  
Never gonna run around and desert you  
Never gonna make you cry never gonna say goodbye  
Never gonna tell a lie and hurt you  
Never gonna give you up never gonna let you down  
Never gonna run around and desert you  
Never gonna make you cry never gonna say goodbye  
Never gonna tell a lie and hurt you

## 常见问题

- 以下命令均等价于 `ls /NJU -l` :
  - `ls -l /NJU`
  - `ls /NJU -l -l -l -l`
  - `ls /NJU -l -ll -lll`
- 以下命令为出错：
  - `ls /NJU -L`
  - `ls /NJU -al`
  - `ls /NJU -lL`
  - `ls /NJU /NJU`
- **直接子目录/文件**即不计算自己的子目录的子目录/文件的数量
- `ls` 中列出多个目录时，其顺序不做规定，每个目录的子文件/目录列表中的各个项的顺序不做规定
- 程序由两个源文件构成，`main.c(cpp)`和`my_print.asm`，其中`main.c(cpp)`是主程序，可以使用C/C++库，但是**输出不能使用库函数，要求在`my_print.asm`中使用汇编编写函数用于输出**
- 要求使用Makefile编译链接项目
  - Windows平台可以使用CMake或其它构建工具
- `.img`文件的名称可以直接在代码中指定（俗称硬编码）
- 要求根据FAT12文件系统格式**直接读取**中的二进制内容，不允许挂载镜像
- FAT12中名称均为大写，**只要求输入为英文大写/数字的指定路径/文件名**（即小写的文件/目录名为不存在），不考虑中文字符，不需要支持长文件名
- 输入指令以回车符号结束，要求在输入`exit`退出程序前，可以多次不断输入
- 程序应该对用户错误的输入做出恰当的提示，**指出错误所在**，不能崩溃

## 评分标准

- 实现基本功能 (`ls`, `ls -l`, `cat`)
- 输出符合要求，对于异常输入能够给出合理提示
- **cat命令要求支持输出超过512字节的文件**

# 提供资料

请在QQ群文件中找到这些我们提供的资料：

- 静态链接、动态链接介绍
- 保护模式
- gcc+nasm联合编译
- FAT12介绍

请提交运行截图、源文件和Makefile文件。

请注意，我们鼓励自己思考和动手实验，如果能够提供自己的思考结果并辅助以相应的实验结果进行说明，在分数评定上会酌情考虑。

如遇到实验相关问题，请在QQ群中提问或向助教发邮件。

## 问题清单

---

在整个实验的过程中，无论是编程还是查资料，请各位同学注意思考以下问题，助教检查时会从中随机抽取数个题目进行提问，根据现场作答给出分数。

## PPT相关内容

1. 什么是实模式，什么是保护模式？
2. 什么是选择子？
3. 什么是描述符？
4. 什么是GDT，什么是LDT？
5. 请分别说明GDTR和LDTR的结构。
6. 请说明GDT直接查找物理地址的具体步骤。
7. 请说明通过LDT查找物理地址的具体步骤。
8. 根目录区大小一定么？扇区号是多少？为什么？
9. 数据区第一个簇号是多少？为什么？
10. FAT表的作用？
11. 解释静态链接的过程。
11. 解释动态链接的过程。

12. 静态链接相关PPT中为什么使用ld链接而不是gcc?
13. linux下可执行文件的虚拟地址空间默认从哪里开始分配。

## 实验相关内容

1. BPB指定字段的含义
2. 如何进入子目录并输出 (说明方法调用)
3. 如何获得指定文件的内容, 即如何获得数据区的内容 (比如使用指针等)
4. 如何进行C代码和汇编之间的参数传递和返回值传递
5. 汇编代码中对I/O的处理方式, 说明指定寄存器所存值的含义