# 操作系统实验（二）问答题参考

南京大学软件学院

2015.5

**实验重点**

本次作业重点：熟悉掌握 Fat12 文件系统，$gcc + nasm$ 联合编译实践以及了解实模式与保护模式的基本内容。

## 1 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。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。

### 1.1 PPT 相关内容

1. 实模式下的寻址方式以及实模式的缺陷

   寻址方式：实模式下，段在内存中固定的位置（物理地址 = 段值 *16 + 偏移）；

   缺陷：通过改变段寄存器的值，我们可以随心所欲的访问内存任何一个单元，而丝毫不受到限制，不能对内存访问加以限制，也就谈不上对系统的保护；内存中每个字节的地址能由不止一个的段基址加偏移表示，比如 04808 能够由 047C:0048,047D:0038,047E:0028 or 047B:0058 表示，分段地址之间的比较将复杂化。

2. 保护模式下的寻址过程：

   - 段寄存器中存储的是什么?GDT 是什么?LDT 是什么? 如何区分 LDT 和 GDT? LDT 和 GDT 的区别是什么? 如何定位到 Descriptor?Descriptor 的内容有哪些?
     - 存储的是选择子，即描述符在描述符表中的位置
     - GDT 是全局描述符表；LDT 是局部描述符表

  - 存储的是选择子，即描述符在描述符表中的位置
  - GDT 是全局描述符表；LDT 是局部描述符表
  - 根据选择子中的第三位决定是 GDT 还是 LDT 区分
  - DT(Local) 与 GDT 相同，但是不是全局的，对于某个进程，它只知道它自己的 LDT。每个进程有自己的 LDT，访问自己的段时从 LDT 查询。进程从 LDTR 寄存器中获得 LDT 的位置，向它发起查询
  - 如果是 GDT，则根据 GDTR 和段寄存器中的内容定位到描述符
  - 如果是 LDT，则根据 LDTR 中的内容（选择子）和 GDTR 中的内容定位到描述符
  - Descriptor 中的内容包括是否在内存中，段的起始地址、界限、属性等内容
  - GDTR 中内容是全局描述符表的位置；LDTR 中存储选择子，用于定位 GDT 中的某个 LDT 描述符，得到 LDT 的地址
  - LDT 存放在 GDT 中的原因是 GDT 表只有一个，是固定的
  - 而 LDT 表每个任务就可以有一个，因此有多个，并且由于任务的个数在不断变化其数量也在不断变化

- GDTR 中的内容是什么?LDTR 中存储的是什么? 为什么 LDT 要放在 GDT 中?
  - 全局描述符表的位置
  - 选择子，用于定位 GDT 中的某个 LDT 描述符，得到 LDT 的地址
  - GDT 表只有一个，是固定的；而 LDT 表每个任务就可以有一个，因此有多个，并且由于任务的个数在不断变化其数量也在不断变化。如果只有一个 LDTR 寄存器显然不能满足多个 LDT 的要求。

3. 选择子的作用：

- 选择子是什么？ 它的值存放在哪里?

  描述符在描述符表中的相对偏移，值存放在段寄存器里。

- 选择子里面的内容有哪些?

  选择子是一个 2 字节的数，共 16 位，最低 2 位表示 RPL（请求特权等级），第 3 位表示查表是利用 GDT（全局描述符表）还是 LDT（局部描述符表）进行，最高 13 位给出了所需的描述符在描述符表中的地址.

- 为什么偏移地址大小是 13 位?

  GDTR 是一个 48 位的寄存器，其中 32 位表示段地址，16 位表示段限（最大 64K，每个描述符 8 字节，故最多有 64K/8=8K 个描述符）13 位正好足够寻址 8K 项。

4. 描述符的作用：

   描述一个段是否在内存中，段的起始地址、界限、属性等内容

5. GDTR/LDTR 的作用：

   - GDTR 的内容是什么?

     全局描述符表的位置

   - LDTR 的内容是什么?

     LDT 的描述符在 GDT 中的相对偏移，根据 GDTR 中的内容即可得到 LDT 的描述符

6. 根目录区大小一定么? 扇区号是多少? 为什么?

   不一定 19 1(引导扇区)+9(FAT1)+9(FAT2) =19

7. 数据区第一个簇号是多少? 为什么?

   第一个簇号为 2，在 1.44M 软盘上，FAT 前三个字节的值必须是固定的，分别是 0xF0、0xFF、0xFF，用于表示这是一个应用在 1.44M 软盘上的 FAT12 文件系统。本来序号为 0 和 1 的 FAT 表项应该对应于簇 0 和簇 1，但是由于这两个表项被设置成了固定值，簇 0 和簇 1 就没有存在的意义了，所以数据区就起始于簇 2。

8. FAT 表的作用?

   记录硬盘中有关文件如何被分散存储在不同扇区的信息（也可以回答为了找到所有的簇 (扇区)）

9. 解释静态链接的过程。

   相似段合并；重定位。

10. 解释动态链接的过程。

    动态链接器自举；装载共享对象；重定位和初始化

11. 静态链接相关 PPT 中为什么使用 ld 链接而不是 gcc。

    使用 ld 进行连接的原因是为了避免 gcc 进行 glibc 的链接

12. linux 下可执行文件的虚拟地址空间默认从哪里开始分配。

    linux 下可执行文件的虚拟空间地址默认从 0x08048000 开始分配

### 1.2 实验相关内容

1、2、3任选一个
4、5选一个
汇编和C代码都要看
传递机制

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
6. 可以要求解释某些看不懂的代码 (我看不懂的话, 你得讲给我听)

## 2 参考资料

1. 《Orange'S: 一个操作系统的实现
2. Introduction to NASM
3. An overview of FAT12
4. Dynamic Linking and Loading