**Question 1:What is the difference between user-level ISA and system-level ISA?**

**权限不同，system-level中有一些特权指令user-level用不了。**

**Question 2: What is memory addressing mode? How many modes are there (please describe them)? Why not just use one?**

**Addressing modes** are an aspect of the [instruction set architecture](https://en.wikipedia.org/wiki/Instruction_set_architecture) in most [central processing unit](https://en.wikipedia.org/wiki/Central_processing_unit) (CPU) designs. The various addressing modes that are defined in a given instruction set architecture define how [machine language](https://en.wikipedia.org/wiki/Machine_code) [instructions](https://en.wikipedia.org/wiki/Instruction_(computer_science)) in that architecture identify the [operand](https://en.wikipedia.org/wiki/Operand)(s) of each instruction. An addressing mode specifies how to calculate the effective [memory address](https://en.wikipedia.org/wiki/Memory_address) of an operand by using information held in [registers](https://en.wikipedia.org/wiki/Processor_register) and/or constants contained within a machine instruction or elsewhere

**一共五个mode, Real-Address Mode , Protected Mode, System Management Mode, IA-32e Mode, Virtual-8086 Mode.**

**为什么不只用一种mode：因为适用的场景不同，real mode不经过MMU，protected mode经过MMU，SMM在系统crash下也可以运行，具有最高的特权，Virtual-8086则是为了兼容， IA-32a扩展内存。这几种mode各有各的优点和适用场景，单用一种mode就可能不会适用所有的场景。**

**Question 3: Use your own word (and figures, if you want) to describe the process from power-on to BIOS end (just before kernel starts)**

What is the usage of "ljmp $(SEG\_KCODE<<3), $start32"?

What is the A20 problem?

加电瞬间强行将CS值置为0XF000，IP为0XFFF0，这样CS:IP就指向0XFFFF0这个位置，这个位置正是BIOS程序的入口地址。BIOS程序被固化在计算机主机板上的一块很小的ROM芯片里。现在CS:IP已经指向了0XFFFF0这个位置，则BIOS开始启动。BIOS启动后就开始执行BIOS代码，先自检，显示显卡，内存的信息，然后枚举本地设备并初始化，在内存中建立中断向量表和中断服务程序，例如0x00000—0x003FF的1KB构建中断向量，0x00400—0x004FF用256字节构建BIOS数据区，0x0E2CE加载了8KB左右的中断服务程序。

接着BIOS会开始准备加载内核，首先，BIOS程序发出int 0x19中断，CPU在中断向量表中找到中断服务程序入口地址后执行。此段代码BIOS设计好，与操作系统无关。此中断程序把软驱的0号磁头对应盘面的0磁道1扇区的内容拷贝至内存0x07C00，此扇区的内容即是linux 0.11的引导程序bootsect，他会把软盘中的操作系统陆续加载进内存。然后操作系统把最开始执行的代码放在0盘面0磁道1扇区， BIOS在接到启动命令后把启动扇区的代码加载到0X07C00。最后加载bootsect的setup程序到内存，这部分的加载也需要利用BIOS的int 0x13中断服务程序。之后内核阶段就开始了。（我参考了<http://blog.csdn.net/gatieme/article/details/50914250>）

"ljmp $(SEG\_KCODE<<3), $start32"：在进入“protected mode”的时候不会让系统进入32位模式，所以调用这段代码将程序从16位转换到32位模式下。

A2 Problem：计算机第一次启动后不久，电源自检（POST）过程中会显示A20错误。 出现此错误消息时， 操作系统尚未加载。当检测到主板上的键盘或键盘控制器出现问题时，POST报告“A20”错误。