

# Checklist

## 1. 解释中断向量

---

- 中断向量把中断 / 异常和它们的处理程序联系起来
- 每种中断都有一个中断向量号，这个向量号通过 IDT 就和中断处理程序对应起来了

## 2. 解释中断类型码

---

把每个中断处理例程进行编号，这个编号就是中断类型码，用来查找中断向量。

## 3. 解释中断向量表

---

存放所有中断向量的地址空间，起始地址为零，最多存放 256 个中断向量，每个中断向量 4 个 bytes，高两个 bytes 存放段描述符，低地址存放偏移。

## 4. 实模式下中断处理程序地址如何得到

---

通过中断号  $n$  从中断向量表中取出中断向量，中断向量中的高两个字节的段描述符中取出段基地址存入 CS，中断向量低两个字节取出偏移量存入 IP，得到中断处理程序的地址。

## 5. 保护模式下中断处理程序的地址如何得到

---

从 IDTR 中获得中断描述符表的起始地址，用中断号  $N \times 8$  得到中断描述符的偏移量，相加得到中断描述符起始地址，根据中断门中的选择子和偏移量得到中断处理程序的首地址（中间还要经过段选择子查段表的过程）

## 6. 中断向量的地址如何得到

---

中断号  $\times 4$

## 7. 实模式下如何根据中断向量的地址得到中断处理程序的地址

---

同 4

## 8. 解释中断描述符

---

8 字节的数据结构，包含用来获取中断处理程序首地址的段选择符和偏移量，还包含其他一些属性位和类型位。

## 9. 保护模式下中断描述符表如何得到

---

由 IDTR 指出中断描述符表的首地址

## 10. 保护模式下中断门如何得到？

---

由 IDTR 获得 IDT 的首地址，由中断号  $\times 8$  加上 IDT 的首地址得到中断门的首地址

## 11. 保护模式下如何根据中断门得到中断处理程序首地址

---

由中断描述符中的段选择符和段偏移量得到（段选择符去查段表得到段基址，和段偏移量相加得到中断处理程序的首地址）

## 12. 中断的分类，举例不同类型的中断

---

从中断源的角度分类，分为：

- 内部异常中断: 由计算机硬件故障或异常引起的中断，比如整除零
- 软中断: 由程序执行了中断指令而引起的中断，比如系统调用
- 外部中断: 由外部设备请求引起的中断，外部中断可分为可屏蔽中断和不可屏蔽中断。

## 13. 中断和异常的区别

---

- 中断指的是由 CPU 以外的事件引起的中断，比如 I/O 中断、时钟中断和控制台中断。
- 异常指的是由 CPU 内部的事件或者程序执行中的事件引起的过程，比如整除零和缺页。

## 14. 实模式和保护模式下中断处理的区别

---

主要的区别就是获取中断处理程序地址的方式，在实模式下，通过中断号查询中断向量表，根据中断向量表中的信息就可以得到中断处理程序的首地址；在保护模式下，IDT 取代了中断向量表的地位，通过中断号查询 IDT 得到段选择子和偏移量，用段选择子查段表获得段描述符得到段基地址和偏移量相加得到中断处理程序的首地址。

## 15. 如何识别组合键，是否有其他方案

---

在 orange's 中，建立扫描码的解析数组记录一个键在组合及非组合状态下的实际值并设置缓冲区；声明了组合键中前者（ctrl、shift 等）相应变量，若出现其 make code 或者 break code，设置其标志位以便与其他非组合键组合在解析数组中找到相应实际值。

## 16. IDT 是什么？有什么用？

---

IDT 是中断描述符表，作用是将每个中断号和一个中断描述符联系起来。

## 17. IDT 中有哪几种描述符

---

中断门描述符，陷阱门描述符，任务门描述符

## 18. 异常的分类

---

- Fault: 一种可以被更正的异常，一旦被更正，可以不失连续性地继续执行，返回地址是产生 fault 的指令
- Trap: 一种发生在 trap 指令执行之后立即被报告的异常，它允许程序或者任务不失连续性地执行。返回地址是产生异常的指令的下一条指令。
- Abort: 不总是报告精确异常发生位置的异常，不允许程序或任务继续执行，而是用来报告严重错误的。

## 19. 用户态和内核态的特权级别分别是多少

---

用户态：3

内核态：0

## 20. 中断向量表中每个中断向量几个字节？里面的结构是什么？

---

每个中断向量四个字节，高两个字节是段描述符，低两个字节是偏移

## 21. 中断执行过程中的共同点、不同点

---

共同点:

- 都是程序执行过程中的强制性转移
- 都是软件或者硬件发生了某种情形而通知处理器的行为

不同点:

- 中断，是 CPU 所具备的功能。通常因为“硬件”而随机发生。异常，是“软件”运行过程中的一种开发过程中没有考虑到的程序错误。
- 中断是 CPU 暂停当前工作，有计划地去处理其他的事情。中断的发生一般是可以预知的，处理的过程也是事先制定好的。处理中断时程序是正常运行的。异常是 CPU 遇到了无法响应的工作，而后进入一种非正常状态。异常的出现表明程序有缺陷。
- 中断是异步的，异常是同步的