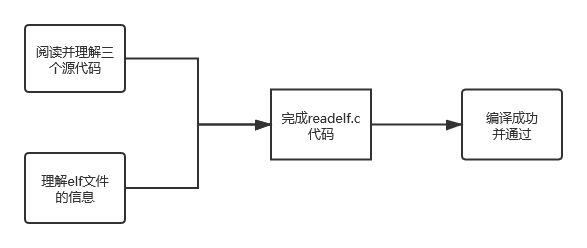
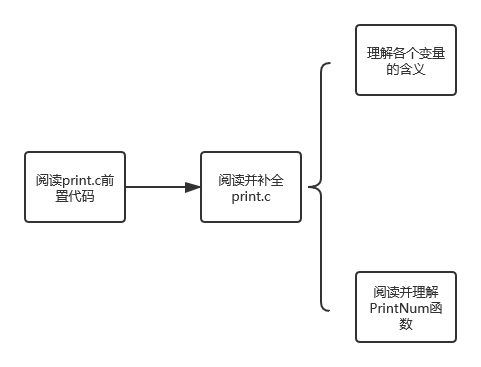
1. 思考题：
2. 因为生成的内核文件的编码方式是大尾端编码，我们的readelf程序只能解析小尾端编码。之后的工具readelf可以解析大尾端编码格式。
3. 内核入口在start.S中，main函数在init目录下的main.c中，通过无条件跳转指令跳到main函数，通过makefile编译链接实现跨文件调用函数。
4. 实验难点



练习1.2是我花费时间最多的练习，阅读三个源代码较为简单，elf文件的格式最初理解的不是很透彻，我感觉最难的就是完成readelf.c代码，主要是对各个elf文件各个变量的类型不是很清晰，还有就是对指针操作有一点生疏，错误频频，浪费了很多时间。





练习1.5不是很难，主要在于理解PrintNum函数，这是这次练习的核心内容，只有理解这个函数才能清楚各个变量的含义，才能在主函数中正确使用。

1. 体会和感想

这次的lab1课下内容总体上说难度不大，个人原因包括指针运用生疏等导致完成的时间比较长，从周一的下午2点左右到晚上12点左右，其中练习1.2花去了近一半的时间，在练习1.5中通过课程群和同学讨论成功修复bug完成所有实验。

通过这次的实验我知道了操作系统的启动过程，包括可执行文件的格式、内核的入口、栈的入口等，也知道了C语言中最常用的printf函数的实现方式，对操作系统有了一点浅显的理解。

1. Lab1-extra

这次的lab1-extra难度不是很大，花费了2个多小时，主要原因是大尾端存储的数据转换成小尾端存储的方法不清楚，花了一些时间去查资料。readelf