《程序员的自我修养》

第一章

1.1

引子

1.2

硬件相关

硬件的关键中央处理器CPU，内存和I/O控制芯片，核心，计算机结构的发展

介绍了对称多处理器（SMP）的优势与适用场景

1.3

软件相关

系统软件，体系结构——层次结构；通信的协议——接口，中间层，虚拟层；介绍了应用程序，运行库，操作系统内核层这几个层级及其所使用的接口。

1.4

介绍了如何挖掘计算机中的资源——CPU、存储器和I/O设备的潜能。

介绍了充分利用CPU的方法发展由多道程序到分时系统再到多任务系统的发展过程，及各种方法提高CPU利用率的方式

操作系统的发展对于程序员工作的简化，不再需要大量的硬件细节。着重介绍了操作系统中的文件系统——其读写方式，穿插介绍了一下硬盘

1.5

介绍了内存分配可能出现的问题，以及使用中间层来解决地址空间隔离问题。

介绍了隔离是如何实现的。

介绍了如何分段是如何实现隔离的，但无法提高内存的使用效率

介绍了分页的实现办法，介绍了页映射的工作原理及功能。

1.6

回顾了线程相关的基础知识。

介绍了线程的概念和组成，与进程的关系。介绍了相比单线程，多线程的优势。

介绍了线程的访问权限，其私有存储空间。

介绍了线程的调度和优先级，在线程数量大于处理器数量时需要考虑。介绍了线程调度的概念和线程调度中线程的三种状态，以及线程状态的切换。介绍了优先级调度和转轮法的概念。介绍了线程按线程优先级的分类，线程优先级的改变方式，“饿死”现象的产生和解决方法。

介绍了抢占的概念，不可抢占线程的执行过程。

介绍了Linux的多线程，创建新任务的方法和使用的代码和具体过程。

介绍了线程安全相关的具体内容。介绍了线程同时访问一个共享数据可能的危害。使用原子操作指令可以解决一部分简单情况下的问题。

紧接着介绍了更加通用的锁，介绍了其实现方式，以及几种不同的锁。

介绍了可重入函数的概念。

介绍了过度优化可能对线程安全造成的危害和解决方法。

介绍了几种线程模型——一对一模型、多对一模型、多对多模型。

1.7

本章小结。

第六章

6.1

了解了程序与进程的区别。学习了是什么决定了虚拟地址空间的大小，如何计算它。了解了虚拟地址空间的分配方式。

学习了PAE的地址扩展方式，介绍了访问AWE访问内存的操作方式。

6.2

了解了动态装入的基本原理。学习了覆盖装入和页映射的实现方法，其中覆盖装入基本已经被淘汰，而页映射是当今的主流。

6.3

站在操作系统的角度了解了一个可执行文件如何被装载，在进程中执行。学习了进程建立的具体过程。介绍了页错误的概念与操作系统处理页错误的过程。

6.4

了解了减少内存浪费的方法，了解了链接视图和执行视图的概念。了解了程序头表的概念以及Elf32\_Phdr结构的各个成员的基本含义。

了解了堆和栈的VMA，知晓了堆的最大申请数量，了解了通过段地址对齐来减少浪费的过程，了解了进程栈的初始化。

6.5  
学习了Linux系统的bash 下输入一个命令执行某个ELF 程序时，Linux 系统如何装载这个ELF文件并且执行。

6.6

了解了Windows PE的装载过程。

6.7

本章小节。