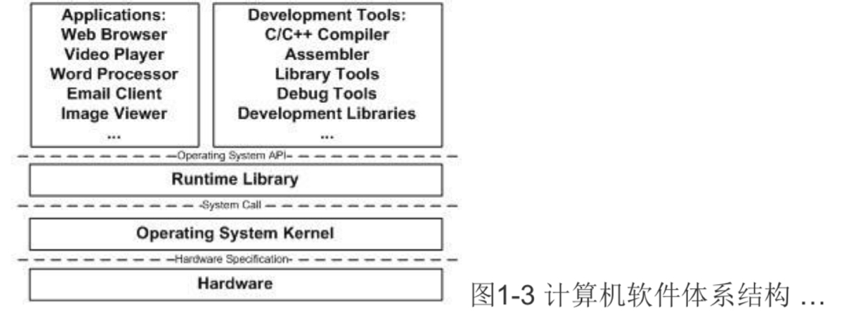
**读《程序员的自我修养》有感**

读此书前，我学过微机原理及其应用这门课程，对计算机硬件有初步但浅薄的了解。在大一上学期时，我学习过C语言，但也只会基础的C语言代码编写，对程序是如何被计算机处理运行一窍不通。当看了此书第一章的前几段话中提出的问题，简单的“Hello world”程序，在深刻的思考中，竟然能衍生处这么多我从没想过的问题，与其说是衍生，不如说是透过表面看到本质的问题。

第一章，用作者的话讲，对整个计算机的软硬件基本结构进行了回顾，包括CPU与外围部件的连接方式、SMP与多核、软硬件层次体系结构、如何充分利用CPU及与系统软件十分相关的设备驱动、操作系统、虚拟空问、物理空问、页映射和线程的基础概念。作者从CPU开始介绍，CPU的工作原理就像一个工厂对产品的加工过程：进入工厂的原料（指令），经过物资分配部门（控制单元）的调度分配，被送往生产线（逻辑运算单元），生产出成品（处理后的数据）后，再储存在仓库（存储器）中，后等着拿到市场上去卖（交由应用程序使用）。接着讲了单一CPU到多CPU的延伸，如SMP，可以提高处理效率。接下来介绍了计算机系统软件体系结构，从上到下分有着严格的层次结构，每个层次相互用“接口”进行通信。



再接着介绍了CPU的抢占式分配方式，使CPU不打盹。

随后是硬件通过驱动程序在操作系统上使用的介绍。

再接着是讲了我比较感兴趣的内容，关于内存的分配问题，读完我了解到计算机通过虚拟地址实现程序运行时的地址隔离，分段解决了地址隔离和地址定位问题，而分页操作使地址使用率显著提高。一共解决了三个核心问题。

最后讲了线程。它是程序执行流的最小单位。它由线程ID，当前指针（PC)，寄存器集合和堆栈组成。一个进程由多个线程组成，一个线程有三个状态：运行、就绪、等待。线程的调度方法包括了优先级调度、轮转法。

至此，第一章的观后思路整理结束。

接着阅读第六章，先分清楚，程序和进程的区别，进程是程序运行时的一个动态过程。程序在执行时，所要的指令和数据必须在内存里才能运行，但若所需的内存大于我们有的内存，将采用动态装入的基本原理进行装载。以前用的是覆盖装入，现在基本被淘汰。当虚拟储存发明后，诞生了页映射，采用装载管理器，按照动态装入原理进行装入过程。目前几乎所有的主流系统都是按照这种方式装载可执行文件的，我们熟悉的Windows文件的装载及Linux对ELF文件的装载都是这样完成的，按着我人操作系统的角度来看可执行文件的装载。在虚拟存储中，现代的硬件MMU都提供地址转换的功能。有了硬件的地址转换和页映射机制，操作系统动态加载可执行文件的方式跟静态加载有了很大的区别。

6.4讲了进程虚存空间分布、6.5讲了Linux内核装载ELF过程，6.6讲了Windows PE的装载，这几个部分内容较为繁琐，就不在此总结。

通过第一章第六章的阅读，我认为我对程序员的认识更深了一步，原来程序员不只是打代码，更要懂得程序运行的深层原理。