

了解低层次的处理器体系结构，那么通过 ORG 和 ORG+ 课程可以达到目的。另一方面，如果你想将当前的计算机组成原理课程转换成 ICS 或者 ICS+ 课程，但是又对突然做这样剧烈的变化感到担心，那么你可以逐步递增转向 ICS 课程。你可以从 OGR 课程开始，它以一种非传统的方式教授传统的问题。一旦你对这些内容感到驾轻就熟了，就可以转到 ORG+，最终转到 ICS。如果学生没有 C 语言的经验（比如他们只用 Java 编写过程序），你可以花几周的时间在 C 语言上，然后再讲述 ORG 或者 ICS 课程的内容。

最后，我们认为 ORG+ 和 SP 课程适合安排为两期（两个小学期或者两个学期）。或者你可以考虑按照一期 ICS 和一期 SP 的方式来教授 ICS+ 课程。

写给指导教师们：经过课堂验证的实验练习

ICS+ 课程在卡内基-梅隆大学得到了学生很高的评价。学生对这门课程的评价，中值分数一般为 5.0/5.0，平均分数一般为 4.6/5.0。学生们说这门课非常有趣，令人兴奋，主要就是因为相关的实验练习。这些实验练习可以从 CS:APP 的主页上获得。下面是本书提供的一些实验的示例。

- 数据实验。这个实验要求学生实现简单的逻辑和算术运算函数，但是只能使用一个非常有限的 C 语言子集。比如，只能用位级操作来计算一个数字的绝对值。这个实验可帮助学生了解 C 语言数据类型的位级表示，以及数据操作的位级行为。
- 二进制炸弹实验。二进制炸弹是一个作为目标代码文件提供给学生的程序。运行时，它提示用户输入 6 个不同的字符串。如果其中的任何一个不正确，炸弹就会“爆炸”，打印出一条错误消息，并且在一个打分服务器上记录事件日志。学生必须通过对程序反汇编和逆向工程来测定应该是哪 6 个串，从而解除各自炸弹的雷管。该实验能教会学生理解汇编语言，并且强制他们学习怎样使用调试器。
- 缓冲区溢出实验。它要求学生通过利用一个缓冲区溢出漏洞，来修改一个二进制可执行文件的运行时行为。这个实验可教会学生栈的原理，并让他们了解写那种易于遭受缓冲区溢出攻击的代码的危险性。
- 体系结构实验。第 4 章的几个家庭作业能够组合成一个实验作业，在实验中，学生修改处理器的 HCL 描述，增加新的指令，修改分支预测策略，或者增加、删除旁路路径和寄存器端口。修改后的处理器能够被模拟，并通过运行自动化测试检测出大多数可能的错误。这个实验使学生能够体验处理器设计中令人激动的部分，而不需要掌握逻辑设计和硬件描述语言的完整知识。
- 性能实验。学生必须优化应用程序的核心函数（比如卷积积分或矩阵转置）的性能。这个实验可非常清晰地表明高速缓存的特性，并带给学生低级程序优化的经验。