

# 《计算机系统》理论课—程序机器级表示基础

## 课堂要点

本次课堂教学主要内容为：高级语言源程序到机器代码的过程及它们之间的差异、数据传送指令及寻址方式、算术操作指令。通过介绍计算机中程序的“机器化”特征，让学生熟悉机器层面信息的表达和使用。本节课可以看做是汇编语言的进一步传授。

1. 一个高级语言源程序需要通过编译才能变成机器能够识别和运行的二进制代码，编译过程中编译器和汇编器及链接器一起，对高级语言代码进行了语法检查、头文件插入、汇编翻译、静态库链接等等操作。

2. 程序在机器中运行时，内存提供代码数据和栈的存储，CPU 负责运算和状态记录。要注意有时候一条 C 代码可能会变成多行机器指令；而有时候多行 C 代码只需要一行机器指令即可执行。

3. 编译获得的汇编代码（.s），反汇编目标文件（.o）获得的汇编代码，以及反汇编可执行目标代码（.out）获得的汇编代码，三者间存在显著区别。

4. 数据传送指令是双操作数指令，依据操作数（源和目的）的不同，可以有多重组合方式：从立即数到寄存器或内存、从寄存器到寄存器或内存，以及从内存到寄存器。没有内存到内存的传送指令（该功能由 DMA 完成）

5. 压栈与弹栈是栈中的数据操作。

压栈：具有默认的目的地（栈顶），因此操作数为要入栈的内容；

弹栈：具有默认的出栈内容（栈顶），因此操作数为目的地。

栈顶指针依据压入或弹出的字节数变化（入栈，ESP 减小；出栈，ESP 增大）。

6. 寻址公式为： $EA=Imm+[Eb+Et*s]$ 。任何一项都可以缺省，要熟悉该公式计算，尤其十六进制的加减法要熟悉操作。

7. 算术操作指令相对较为简单，需要注意的是每个操作数的含义，以及特殊运算指令的默认寄存器内容（例如 `imul1 S` 的一个乘数必须提前默认放入 `%eax`）