# 关于编译过程的理解

在VC++6.0中，在编译时会在当前路径下生成debug文件夹，用于存放编译过程中生成的中间文件。

（1）预处理：首先会生成i文件，这是预处理后的文件，主要负责展开所有宏定义、处理预编译指令、删除注释、添加行号和文件标示，以便编译时禅师错误警告显示行号。

（2）编译：将预处理完的i文件进行一系列词法分析、语法分析、语义分析及优化后生成相应的汇编代码文件（s文件），这是整个程序构建中最核心的部分，也是最复杂的部分。

（3）汇编：将s文件汇编为机器代码文件，也就是二进制目标文件，也叫o或者obj文件。

（4）链接：动态链接/静态链接

# 关于C语言与内存应用的理解

1、系统在利用栈这种逻辑结构时，需要定义一些额外的数据对栈进行物理化。其中，栈帧也叫过程活动记录，是编译器用来实现函数调用的一种数据结构。C语言中，每个栈帧对应着一个未运行完的函数。从逻辑上讲，栈帧就是一个函数调用环境：函数调用框架、参数参数、函数的局部变量、函数执行完后返回到哪里等等。

2、对于x86架构的CPU

（1）寄存器ebp（base pointer ）可称为“帧指针”或“基址指针。  
（2）寄存器esp（stack pointer）可称为“ 栈指针”。

3、在VC++下，连接器对控制台程序设置的入口函数是 \_mainCRTStartup，mainCRTStartup 再调用main 函数；所以当我们操作时，首先会给mainCRTStartup（）函数开辟一段空间，然后esp和ebp在他们所在的位置。

4、函数执行过程：

（1）push ebp；将ebp的地址压入栈中

（2）mov ebp，esp；使ebp=esp，即ebp指向栈顶位置

（3）sub esp，0E4h；为main开辟存储空间

（4）初始化开辟空间

（5）实参入栈

（6）调用sub函数，形参入栈

（7）call指令

（8）执行sub函数直到调用结束，释放栈帧

（9）释放main函数，执行ret指令