**C语言编译过程**

C语言编译过程分成四个步骤：

1. 由.c文件到.i文件，这个过程叫预处理
2. 由.i文件到.s文件，这个过程叫编译
3. 由.s文件到.o文件，这个过程叫汇编
4. 由.o文件到可执行文件，这个过程叫链接

具体步骤为：

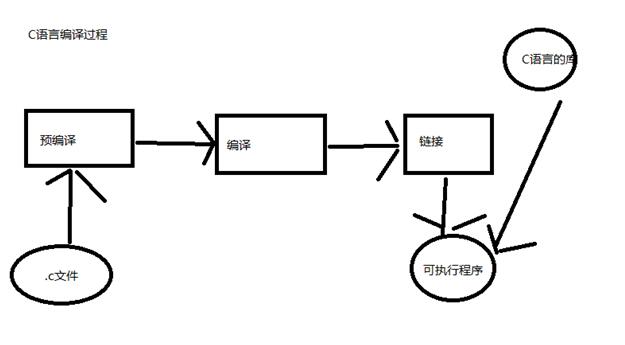
预处理： **gcc -E -I./inc test.c -o test.i**

编译：  **gcc -S hello.c -o a.s**

汇编： **gcc -c test.s -o test.o**

链接： **gcc xxx.o xxx1.o xxx2.o -o link**





【第一步】预处理

预处理过程实质上是处理“#”，将#include包含的头文件直接拷贝到hell.c当中；将#define定义的宏进行替换，同时将代码中没用的注释部分删除等。

具体做的事儿如下：

（1）将所有的#define删除，并且展开所有的宏定义。说白了就是字符替换。

（2）处理所有的条件编译指令，#ifdef #ifndef #endif等，就是带#的那些。

（3）处理#include，将#include指向的文件插入到该行处。

（4）删除所有注释。

（5）添加行号和文件标示，这样的在调试和编译出错的时候才知道是是哪个文件的哪一行。

1. 保留#pragma编译器指令，因为编译器需要使用它们。

gcc的预处理是预处理器cpp来完成的，你可以通过如下命令对test.c进行预处理：

**gcc -E -I./inc test.c -o test.i**

或者直接调用cpp命令

**$ cpp test.c -I./inc -o test.i**

上述命令中-E是让编译器在预处理之后就退出，不进行后续编译过程；-I指定头文件目录，这里指定的是我们自定义的头文件目录；-o指定输出文件名。

【第二步】编译

编译的过程实质上是把高级语言翻译成机器语言的过程，即对a.c做了这些事儿。

（1）词法分析

（2）语法分析

（3）语义分析

（4）优化后生成相应的汇编代码

从 高级语言->汇编语言->机器语言（二进制）

**gcc -S hello.c -o a.s** 可以生成汇编代码

上述命令中**-S**让编译器在编译之后停止，不进行后续过程。编译过程完成后，将生成程序的汇编代码test.s，这也是文本文件。

【第三步】汇编

汇编过程将上一步的汇编代码转换成机器码(machine code)，这一步产生的文件叫做目标文件，是二进制格式。

gcc汇编过程通过as命令完成：

**$ as test.s -o test.o**

等价于：

**gcc -c test.s -o test.o**

这一步会为每一个源文件产生一个目标文件。

【第四步】链接

就像刚才的hello.c它使用到了C标准库的东西“printf”，但是编译过程只是把源文件翻译成二进制而已，这个二进制还不能直接执行，这个时候就需要做一个动作，将翻译成的二进制与需要用到库绑定在一块。

打个比方编译的过程就向你对你老婆说，我要吃雪糕。你只是给你老婆发出了你要吃雪糕的诉求而已，但是雪糕还没有到。绑定就是说你要吃的雪糕你的老婆已经给你买了，你可以happy。

**gcc xxx.o xxx1.o xxx2.o -o link**