

初步了解编译过程

预处理

C语言的预处理过程将源代码翻译为一个ASCII中间文件,它是一个经过修改后的源代码,预处理有文件包含、宏展开、条件编译这三个功能。文件包含是将类似“#include”这样的包含声明扩展为程序正文。宏展开是将通过“#define”定义的宏展开为相应的值。条件编译是根据“#if”等编译命令,将源程序中某些部分包含进来或忽略过去。在GCC中,生成预处理后的文件的命令为:

```
gcc -E -o test.i test.c
```

在Clang中,生成预处理后的文件的命令与GCC是相同的:

```
clang -E -o test.i test.c
```

编译

这里的编译包括两个过程,首先编译器将预处理后的源代码编译为汇编码,然后汇编器将汇编码转换为机器语言,生成可重定位目标文件。GCC中编译的命令如下:

```
gcc -S test.c //得到汇编文件test.s
gcc -o test.o main.s //得到可重定位目标文件test.o
```

Clang中编译的命令与GCC中相同

连接

连接过程将多个可重定位目标文件组成一个可执行目标文件,以便程序能够装入内存运行。连接分为静态连接和动态连接,其中静态连接的连接过程发生在生成可执行文件时,动态连接的过程发生在程序装入内存将要执行时。连接器主要完成两个任务,一个是符号连接,识别各个目标模块中的符号,为每一个符号引用确定它的符号定义;另一个是重定位,将符号定义关联到适当的内存位置,并使它们的引用指向那个位置。GCC中连接的命令如下:

```
gcc -o test test.o //将可重定位目标文件连接为可执行文件
gcc -o test test.c -l libfoo //连接静态库
```

Clang中命令相同。