**一，开始**

头文件和源文件？

通常，在一个 C++ 程序中，只包含两类文件：.cpp 文件和 .h 文件。

C++ 语言支持"分别编译"。一个程序所有的内容，可以分成不同的部分分别放在不同的 .cpp 文件里。比如，在文件 a.cpp 中定义了一个全局函数 "void a(){}"，而在文件 b.cpp 中需要调用这个函数。即使这样，文件 a.cpp 和文件 b.cpp 并不需要相互知道对方的存在，而是可以分别地对它们进行编译，编译成目标文件之后再链接，整个程序就可以运行了。

这是怎么实现的呢？在文件 b.cpp 中，在调用 "void a()" 函数之前，先声明一下这个函数 "voida();"，就可以了。这是因为编译器在编译 b.cpp 的时候会生成一个符号表，像 "void a()" 这样的看不到定义的符号，就会被存放在这个表中。再进行链接的时候，编译器就会在别的目标文件中去寻找这个符号的定义。一旦找到了，程序也就可以顺利地生成了。

注意两个概念："定义"和"声明"。简单地说，"定义"就是把一个符号完完整整地描述出来。而"声明"则只是声明这个符号的存在，即告诉编译器，这个符号是在其他文件中定义的，我这里先用着，你链接的时候再到别的地方去找找看它到底是什么吧。需要注意的是，一个符号，在整个程序中可以被声明多次，但却要且仅要被定义一次。

如果有一个很常用的函数 "void f() {}"，在整个程序中的许多 .cpp 文件中都会被调用，那么，我们就只需要在一个文件中定义这个函数，而在其他的文件中声明这个函数就可以了。一个函数还好对付，声明起来也就一句话。但是，如果函数多了，比如是一大堆的数学函数，有好几百个，那怎么办？