

GCC 和 cmake的关系？



辉常哥

充电2分钟，装逼5小时

- 1.gcc是GNU Compiler Collection（就是GNU编译器套件），也可以简单认为是编译器，它可以编译很多种编程语言（括C、C++、Objective-C、Fortran、Java等等）。
- 2.当你的程序只有一个源文件时，直接就可以用gcc命令编译它。
- 3.但是当你的程序包含很多个源文件时，用gcc命令逐个去编译时，你就很容易混乱而且工作量大
- 4.所以出现了make工具
make工具可以看成是一个智能的批处理工具，它本身并没有编译和链接的功能，而是用类似于批处理的方式—通过调用makefile文件中用户指定的命令来进行编译和链接的。
- 5.makefile是什么？简单的说就像一首歌的乐谱，make工具就像指挥家，指挥家根据乐谱指挥整个乐团怎么样演奏，make工具就根据makefile中的命令进行编译和链接的。
- 6.makefile命令中就包含了调用gcc（也可以是别的编译器）去编译某个源文件的命令。
- 7.makefile在一些简单的工程完全可以人工手下，但是当工程非常大的时候，手写makefile也是非常麻烦的，如果换了个平台makefile又要重新修改。
- 8.这时候就出现了Cmake这个工具，cmake就可以更加简单的生成makefile文件给上面那个make用。当然cmake还有其他功能，就是可以跨平台生成对应平台能用的makefile，你不用再自己去修改了。
- 9.可是cmake根据什么生成makefile呢？它又要根据一个叫CMakeLists.txt文件（学名：组态档）去生成makefile。
- 10.到最后CMakeLists.txt文件谁写啊？亲，是你自己手写的。
- 11.当然如果你用IDE，类似VS这些一般它都能帮你弄好了，你只需要按一下那个三角形

编辑于 2016-03-08

▲ 赞同 259 ▼ ● 评论 12 ★

收起 ^

App 内打开
打开 App，查看更多回答