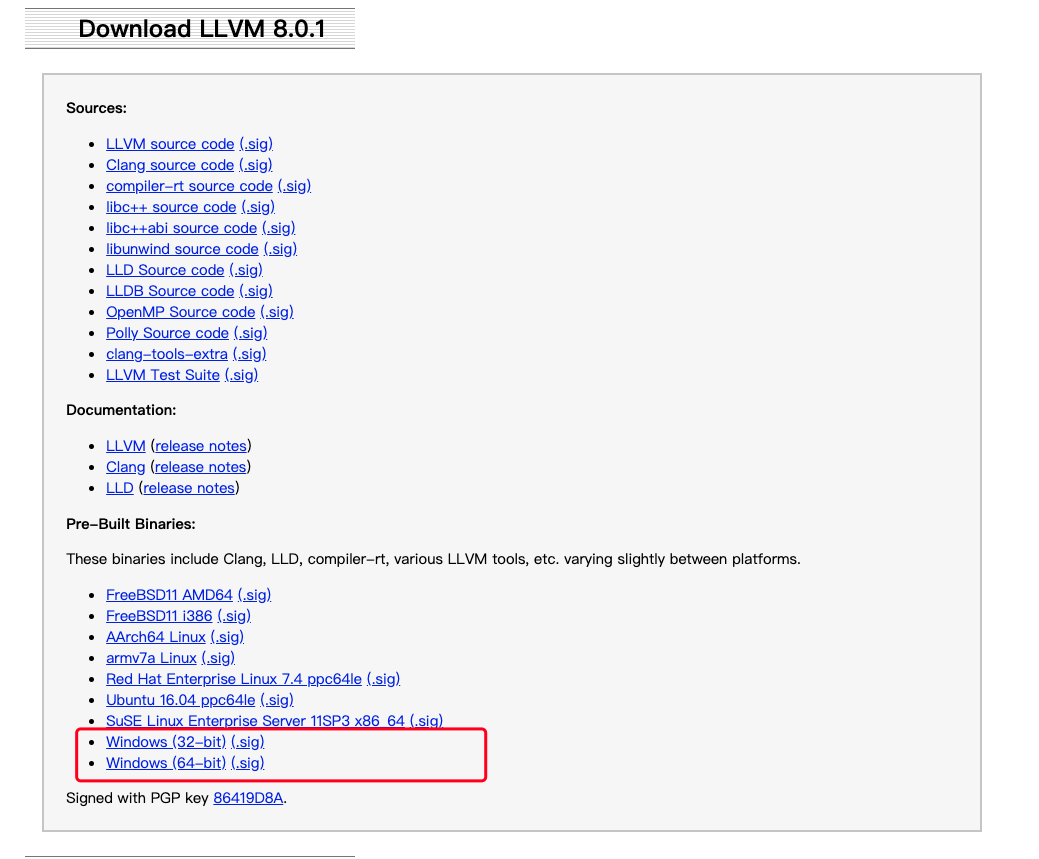
本学期建议的集成开发环境是VS Community2019和Codeblocks 17.12，需要配置clang 8.0.1编译器，产生的MIPS汇编在Mars模拟器上运行。

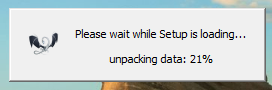
集成开发环境主要是用于调试，尤其是在上机考试时进行现场调试，VS Community2019 和Codeblocks 17.12任选一种即可。

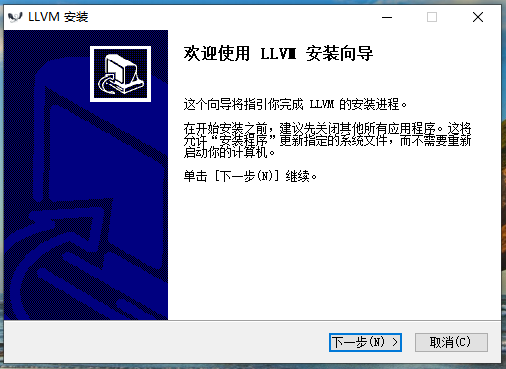
**开发环境安装方式**

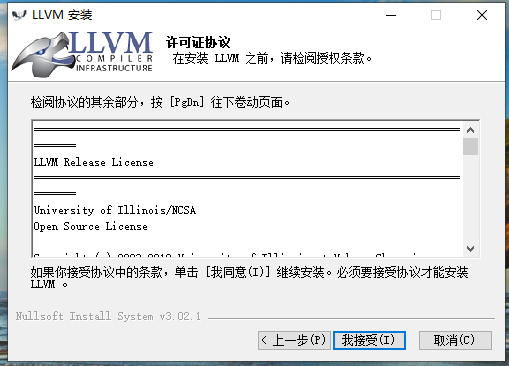
两种开发环境都需要安装LLVM Clang，在 <https://releases.llvm.org/download.html> 页面下下载LLVM 8.0.1版本的安装包,根据自己的操作系统进行选择，图中标出的是Windows平台的安装包。

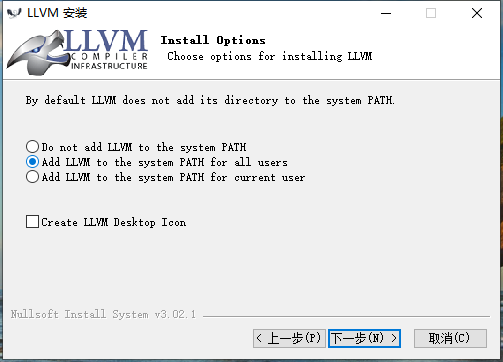


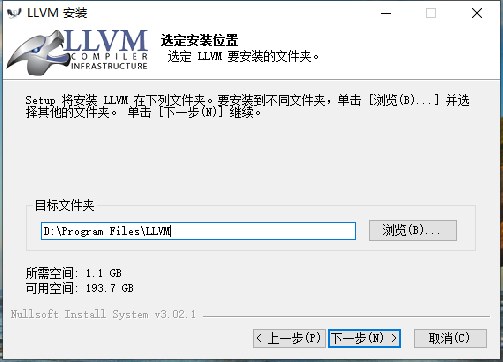
下载好后进行安装：

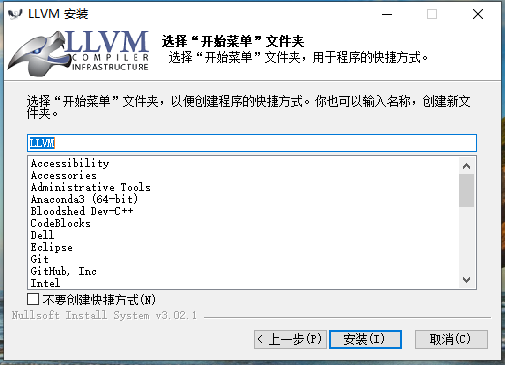












安装完LLVM后，选择以下其中一种开发环境进行配置。

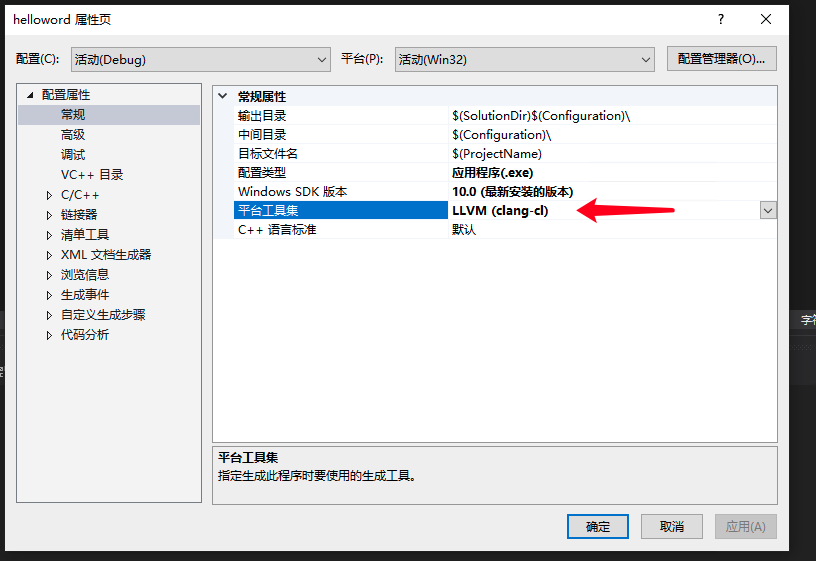
1. **使用VS Community2019 进行开发**
2. 下载并安装VS Community2019
3. 选择C/C++开发，除了保持默认选项外还需要勾选即“适用于 Windows 的 C++ Clang 工具”和“MSVC v141”



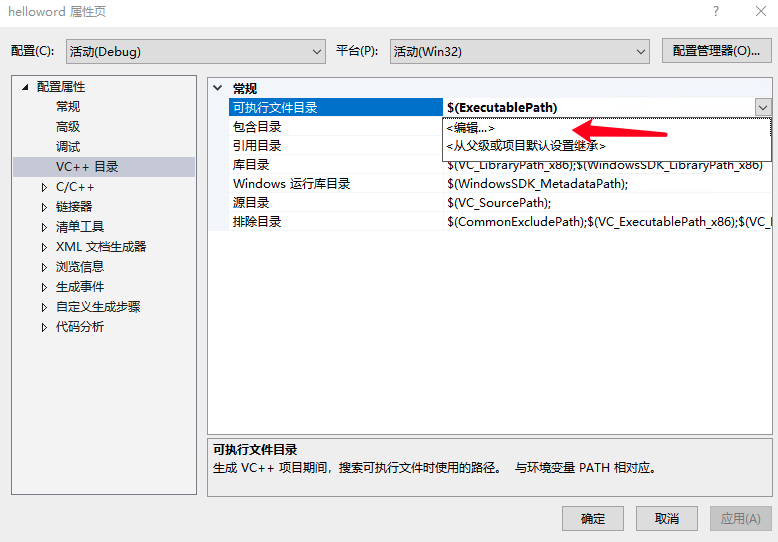
1. 打开VS安装位置下的VC\Tools\MSVC 文件夹（使用默认安装位置的话为 C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\VC\Tools\MSVC）

先保存下以 14.2 开头的文件夹的名字，然后将此文件夹命名成其他，最后将以 14.1 开头的文件夹重命名成之前保存的以 14.2 开头的文件夹的名字。

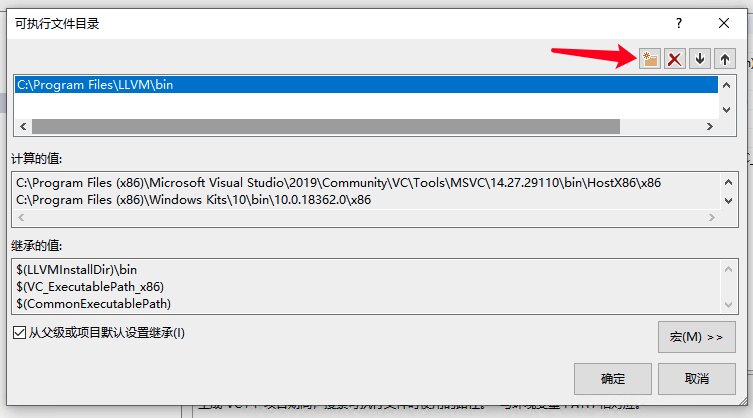
1. 设置VS项目使用Clang：在菜单“项目”-“项目属性”下将“平台工具集”选择为LLVM



然后在“VC++”目录下编辑“可执行文件目录”：



添加LLVM安装目录下的bin文件夹

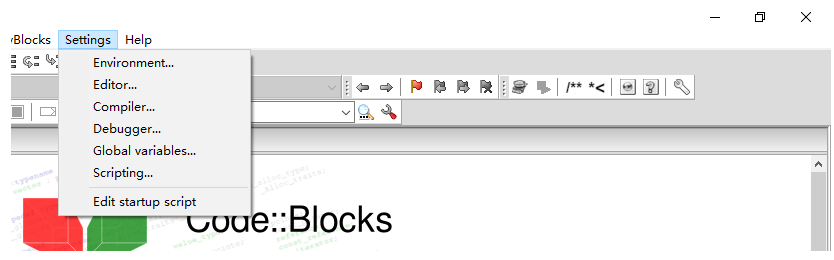


注：在VS中建立的每个工程都需要配置步骤4

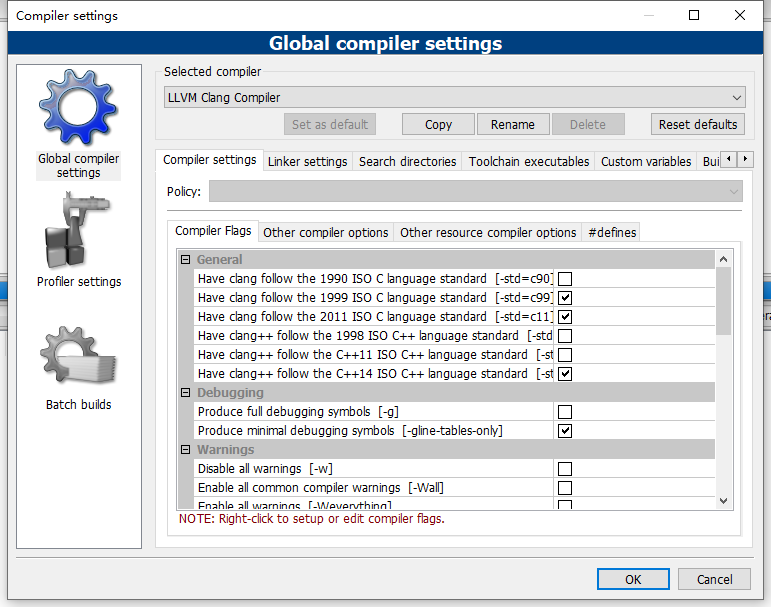
1. **使用Code Blocks进行开发**
2. 下载mingw64的8.1.0版本并解压，并把其下的bin目录加入系统变量path中。
3. 下载并安装Code Blocks 17.12

3. 指定Code Blocks的编译器为LLVM Clang

选择Settings -> Complier -> Selected Complier选择LLVM Clang



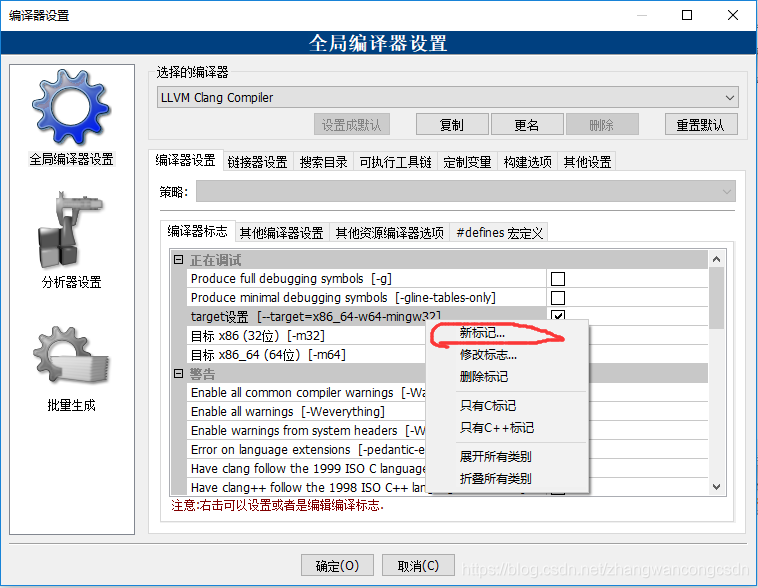
注意Set as default



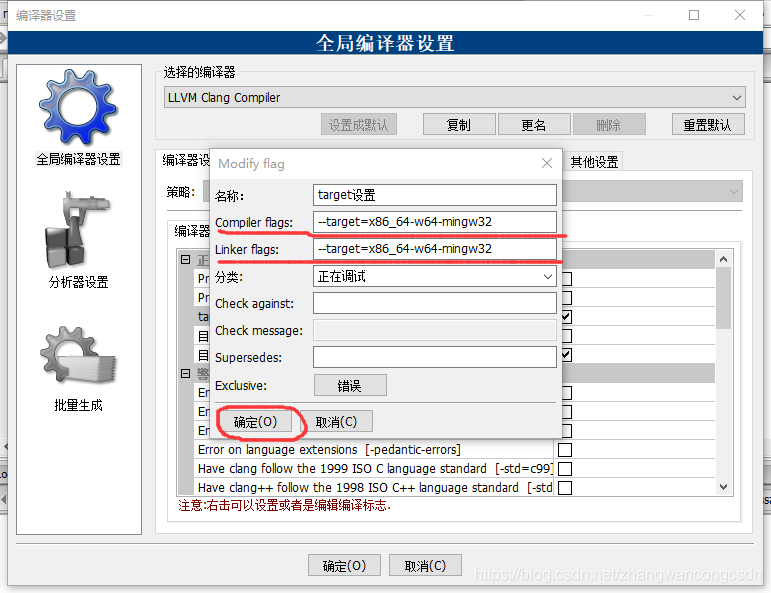
指定LLVM 的位置：



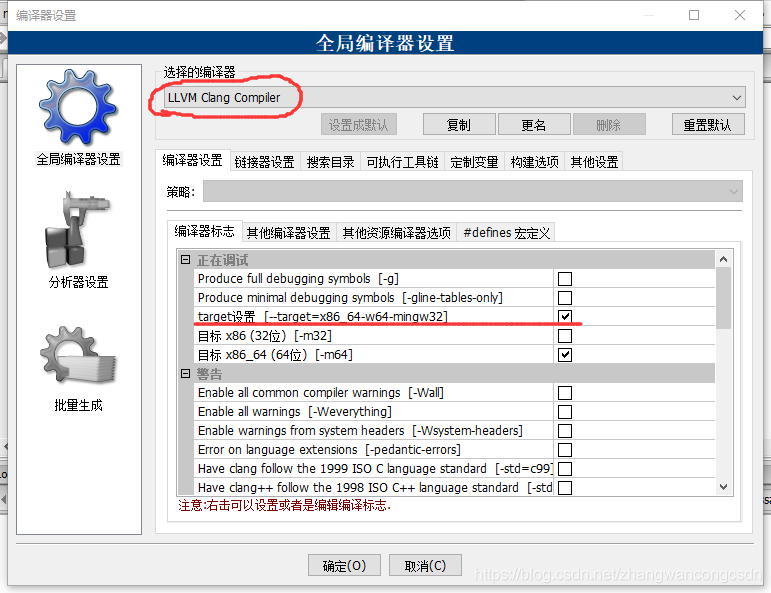
为Clang设置target标志：在编译器设置栏右击任意一个选项，弹出菜单中选择“新标记”



把编译器选项和连接器选项都加上 *--target=x86\_64-w64-mingw32* ，单击确定：



并且勾选这个新添加的选项:



保存设置后，重启Code Blocks即可使用Clang编译。

参见:

https://blog.csdn.net/zhangwancongcsdn/article/details/100918915

**相关文件下载地址：**

VS 2019：<https://visualstudio.microsoft.com/thank-you-downloading-visual-studio/?sku=Community&rel=16>

LLVM Clang：<http://releases.llvm.org/download.html>（注意下载LLVM 8.0.1）

Code Blocks 17.12：https://sourceforge.net/projects/codeblocks/files/Binaries/17.12/

Mingw64 8.1.0：https://bhpan.buaa.edu.cn:443/link/2DC0CD4EB7A05A53B1F5CBC95FBA86FA