**C++实验报告**

1. vs2022的安装

到微软官方网站去下载vs2022的安装程序VisualStudioSetup.exe。之后根据提示进行安装



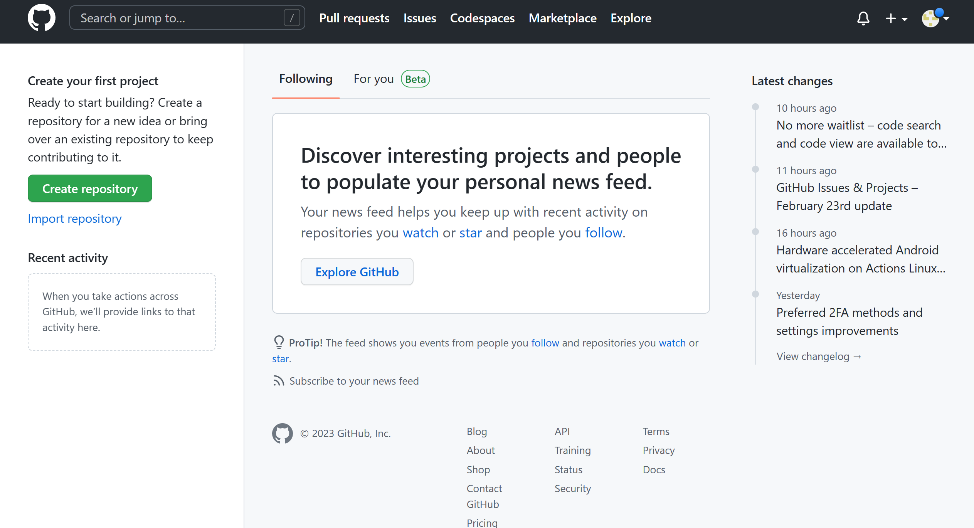
注意点：1.安装时自动安装的并不是professional版本，需要点击下载后结束任务，重新选择professional版本）

2．在选择语言时需要同时选择中文和英文，否则安装后可能会出现某些功能不可用的情况。

二、github的注册

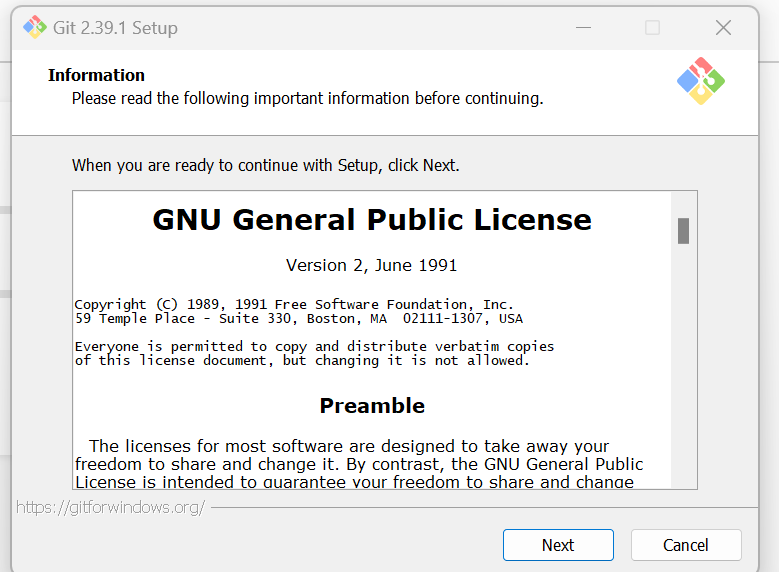
GitHub是全球最大的程序员交流网站，上面拥有数不尽的优秀程序资源。为下一步git的使用以及远程仓库的安装，我们需要出则好GitHub账号。

安装好之后，创建自己的个人仓库。

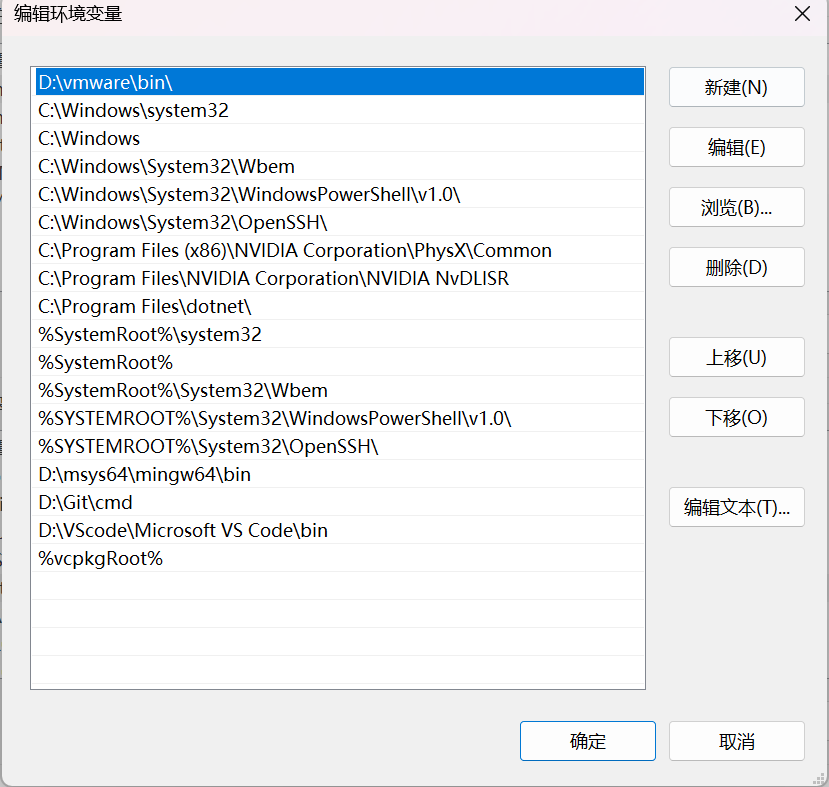


三、git的安装与使用

去git官方网站<https://git-scm.com/下载安装程序，选择windows 64>位版本。



接下来执行程序，安装git到指定目录。然后设置系统环境变量。（会发现已经自动配置好）



Git有什么用？我们为什么要使用他呢？简单来说，git最重要的一个作用就是，可以将我们的本地仓库与云端（如github、gitee）等云端仓库进行连接，可以自动对我们对仓库内文件进行的修改进行备份，快照。能够有效的防止工作内容的丢失，同时也可以方便的在云端上下载到任何我们所需要的提交过的文件版本。

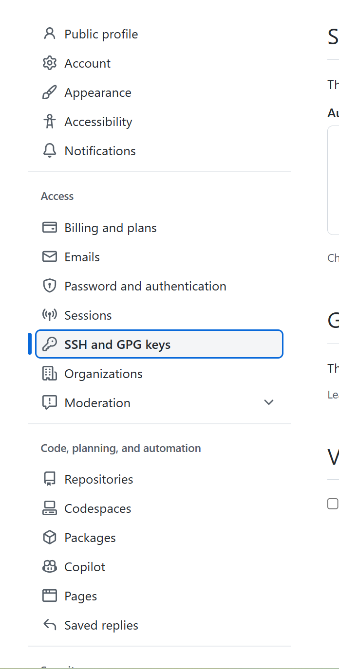
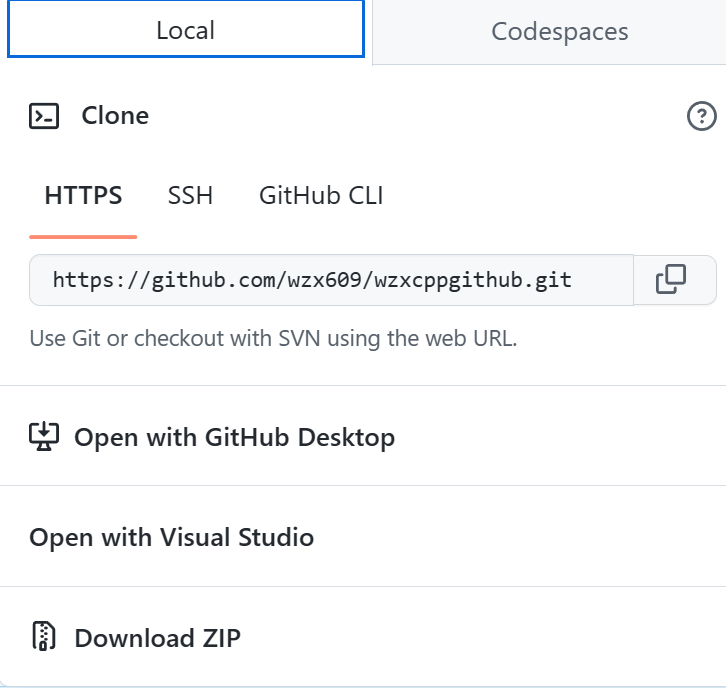
接下来我们就需要将本地仓库与云端进行连接。首先，在目录下创建一个文件夹，并且命名（最好与GitHub内仓库同名，方便未来查找），右键弹出菜单，选择git bash



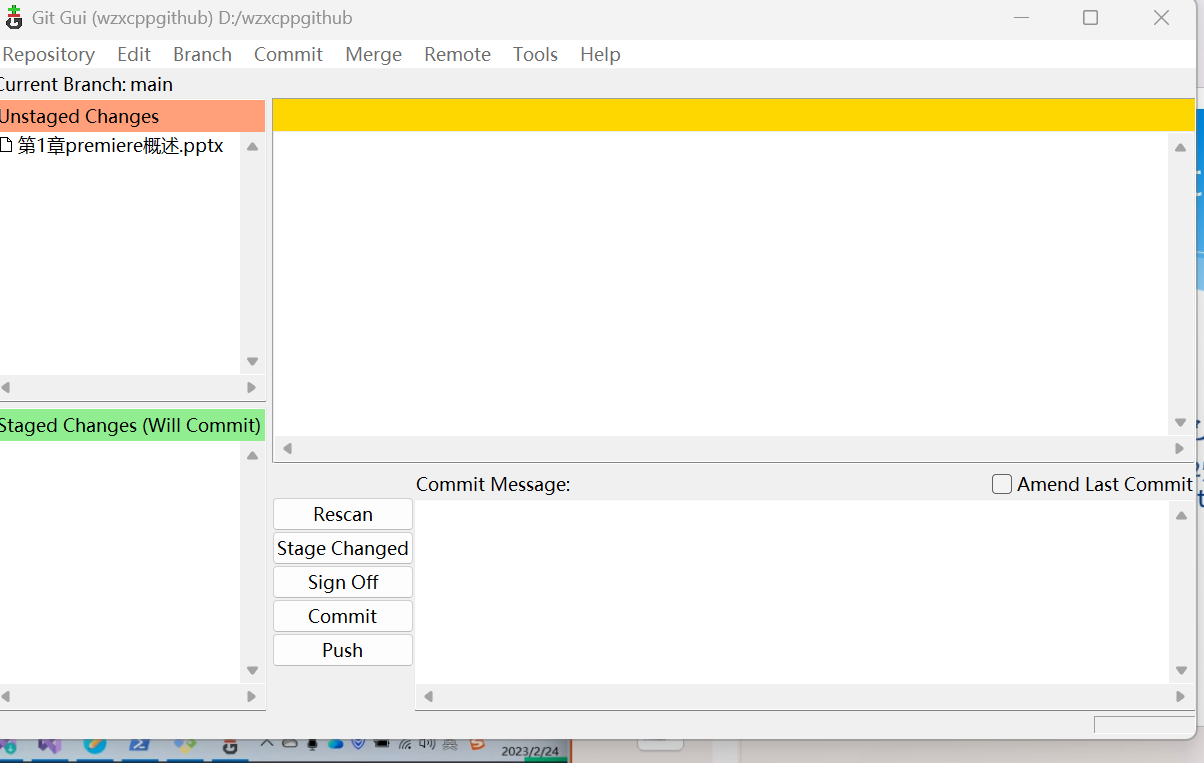
输入命令git init，之后执行用户民名与密码命令git config --global user.email“电子邮箱”，git config --global user.name“用户名。执行成功则会生成密钥对（注意点：生成密钥对时，需要连按两下回车），此时c盘用户目录下会生成一个.ssh文件，用记事本打开名为id\_rsa.pub文件，复制其中内容（这就是生成的密钥）。

  
接下来，到GitHub点击个人头像，选择setting选项，在右侧选则SSH选项，选择NEW SSH KEY，之后输入自己复制的密钥，确定，云端仓库及创建成功 ，接下来我们可以输入$ ssh -T [git@github.com](mailto:git@github.com)，来检测密钥是否有用，若出现ermanently added the ED25519 host key for IP address '20.205.243.166' to the list of known hosts.

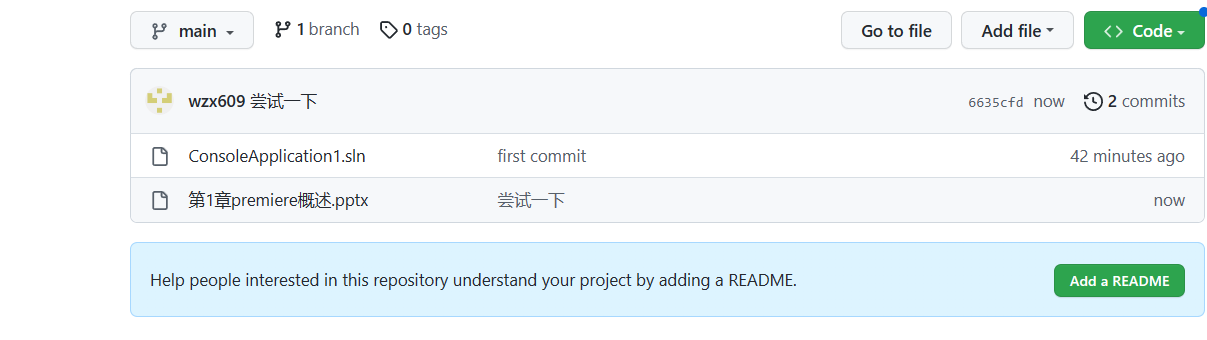
Hi\*\*\*\*!则说明已经连接成功，之后登录github，打开仓库，复制链接。

*   
  在命令窗口，复制刚才的链接git remote add origin https://github.com/用户名/仓库名.git，执行后便建立了远程与本地的连接。

接下来我们可以复制几个文件到该仓库进行尝试，添加所有文件到项目中$ git add .尝试提交所有文件，$ git commit -m "备注信息" -a，$ git branch -M main推送到远程仓库，$ git push -u origin main第一次需要参数-u，以后就不需要了，直接执行git push origin main就可以了。我们也可以右键仓库，选择git GUI here



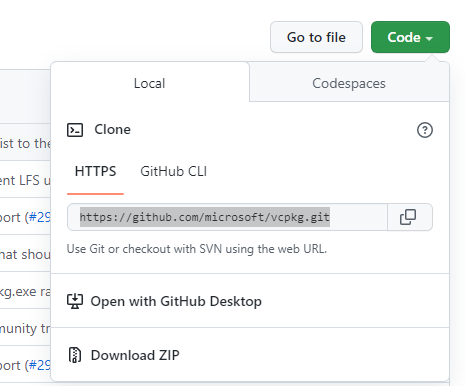
点击stage changed，添加备注（注意点：向github提交任何文件时都必须要添加备注），之后点击commit，之后出现success及说明提交成功，可去GitHub上查看



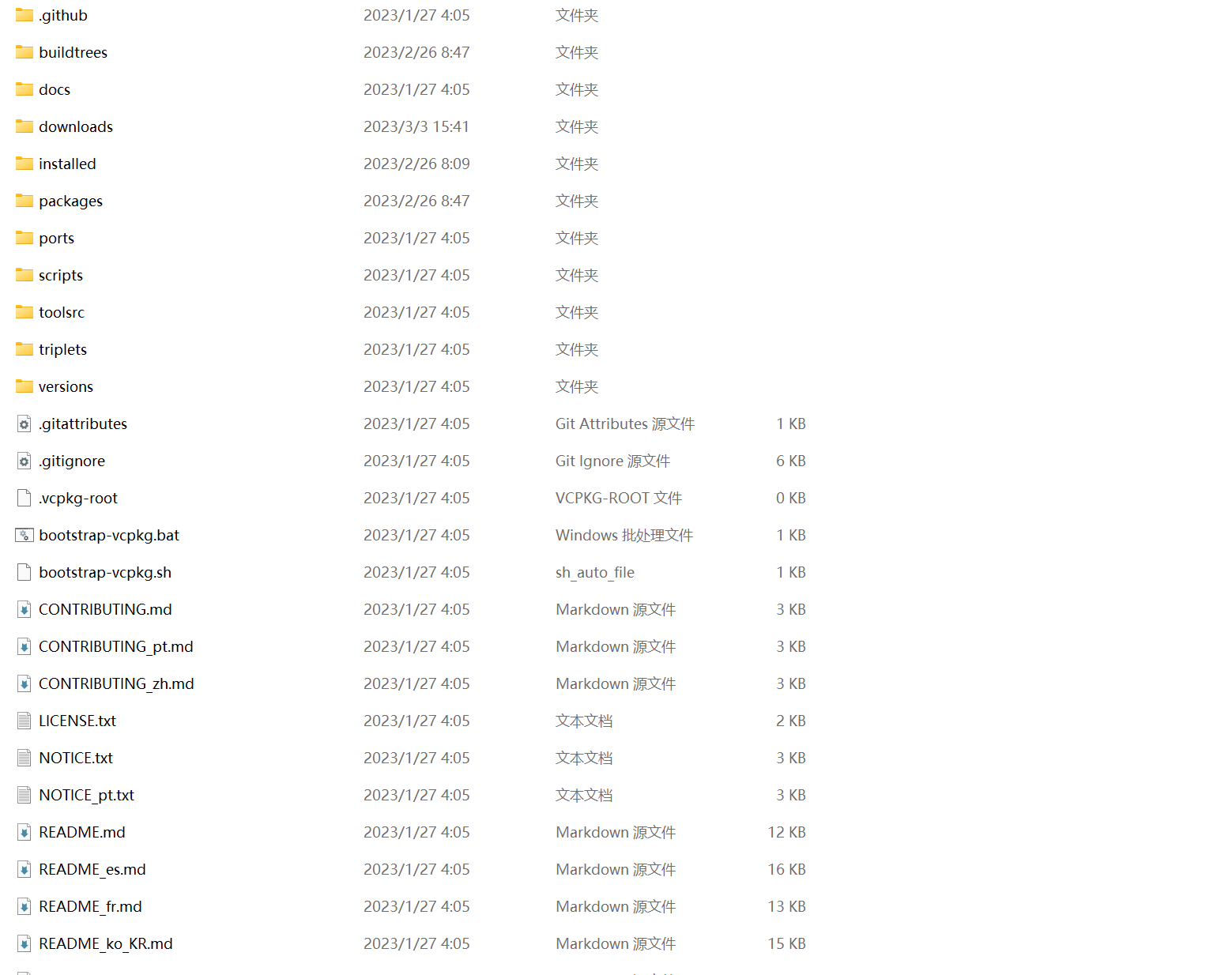
1. vcpkg的安装

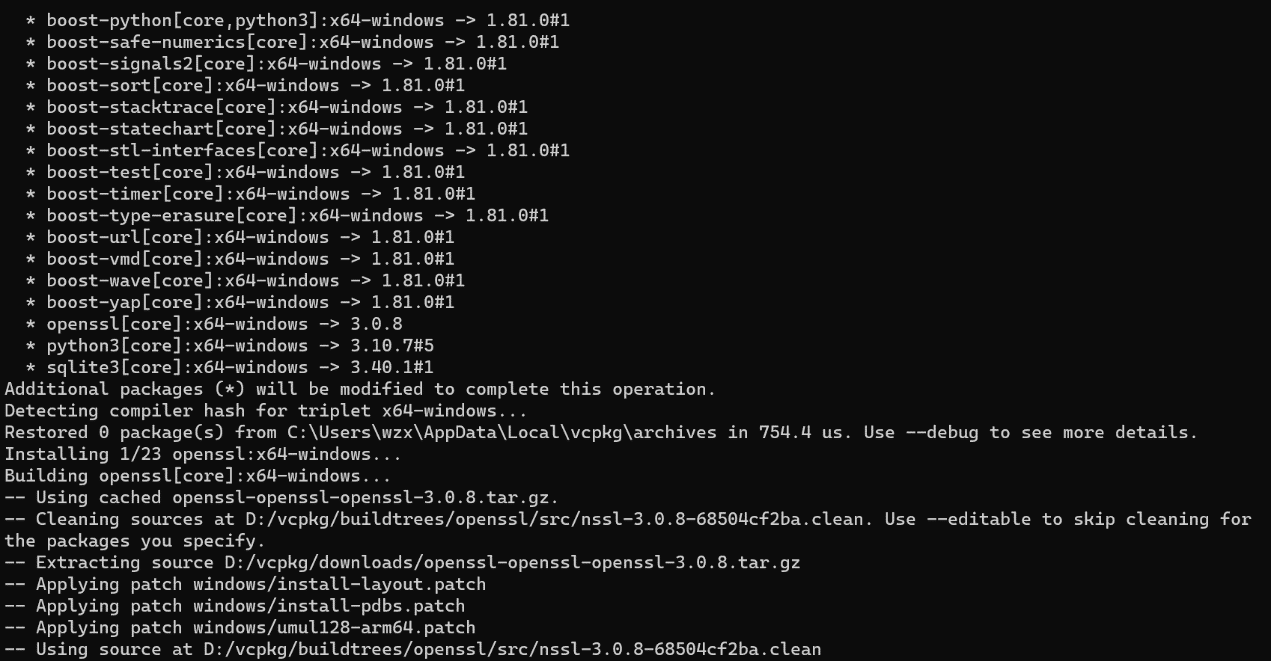
在浏览器中输入<https://github.com/microsoft/vcpkg>

点击code复制地址

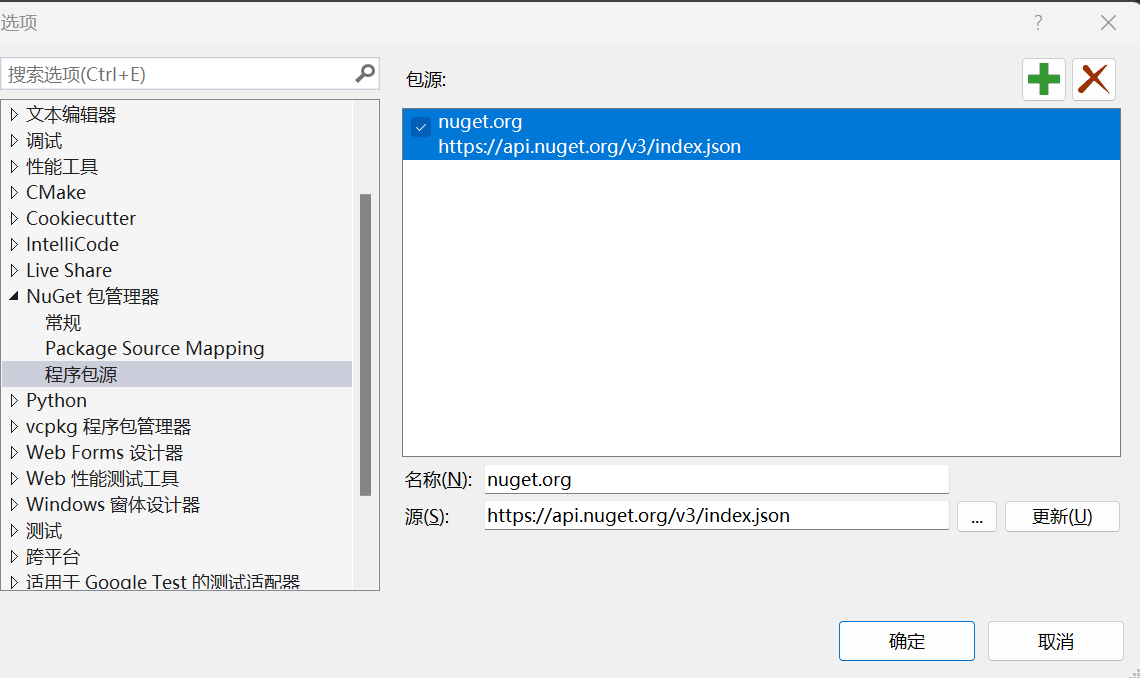


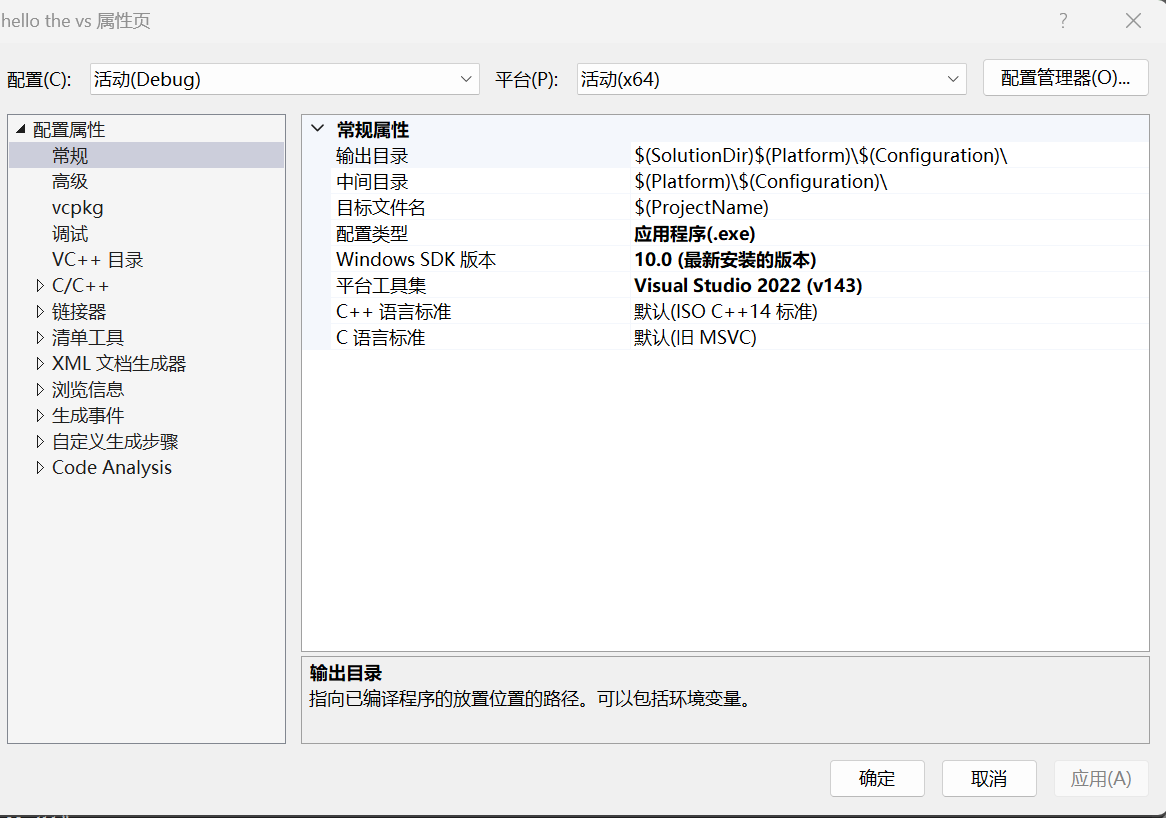
之后，打开D盘，右键空白处，打开命令行，输入git clone <https://github.com/microsoft/vcpkg>，回车执行命令，之后即开始下载vapkg程序。

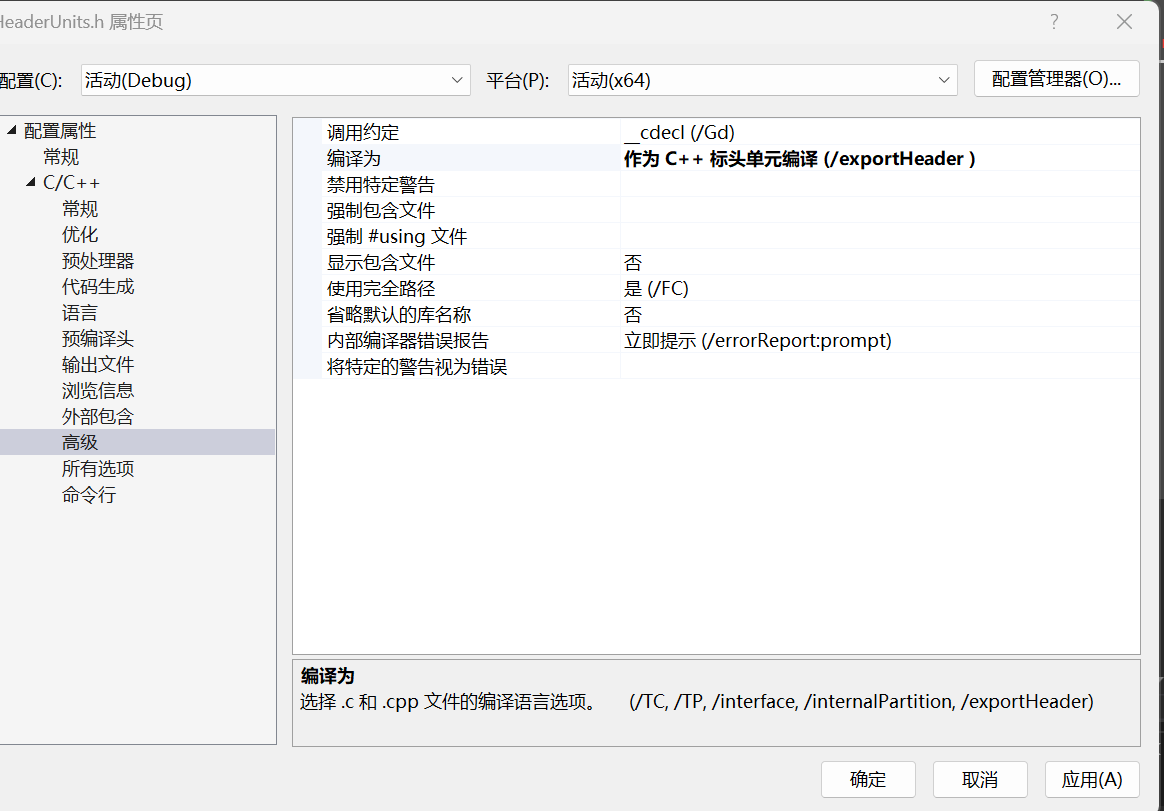




接下来，win+r输入cmd，进入中端，切换到d:vcpkg目录，输入bootstrap-vcpkg.bat，这样就会产生vcpkg.exe程序，之后我们就可以下载boost，执行vcpkg install boost:x64-windows，即可开始下载boost第三方库。之后，进行一系列配置，我们就可以在vs2022中使用vcpkg，还可以配置c++20。





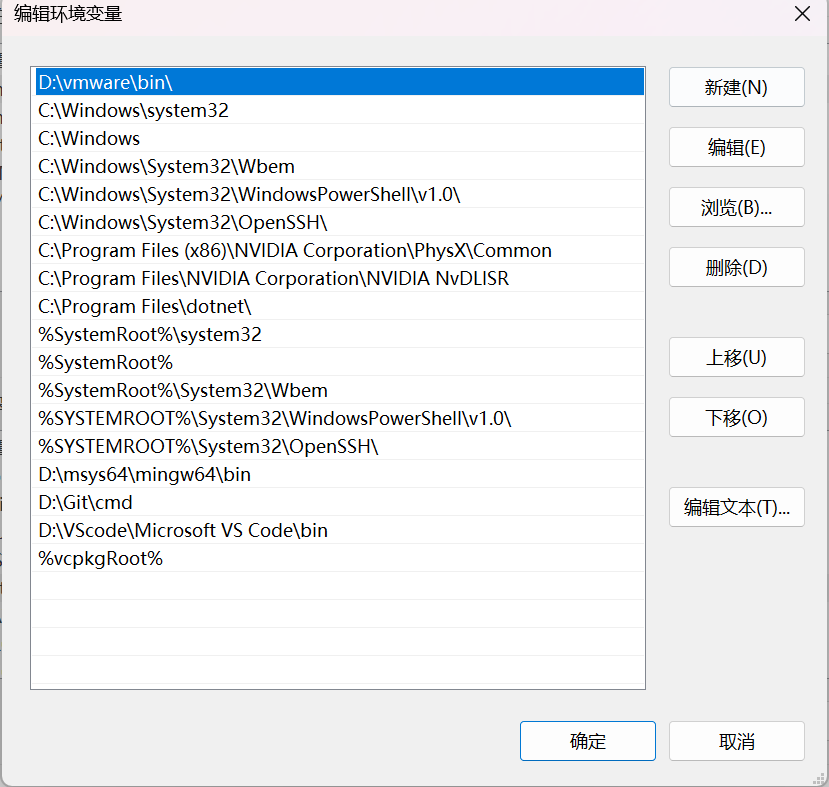


1. vscode安装与使用
2. 安装mingw

Vscode不同于vs，其是一款完全免费且开源的软件，但缺点是我们需要自行进行一些列环境配置。

Vscode本质上只是一个代码编辑器，并不具备编译与调试功能，所以在这里我们选择mingw作为编译与调试工具。去https://www.msys2.org/网站下载mingw-w64，

在打开的网页中点击msys2-x86\_64-20230127.exe，之后执行安装命令即可。接下来要进行系统环境变量配制，



（在path中添加一条环境变量，切记不要更改其他变量，否则会出现严重问题）之后，输入<https://packages.msys2.org/queue>，在顶部的搜索框中输入gcc，search，点击mingw-w64-gcc，选择mingw-w64-x86\_64-gcc，复制链接。之后，打开安装好的mingw程序（蓝色，一定不要点错），分别执行以下三条指令：

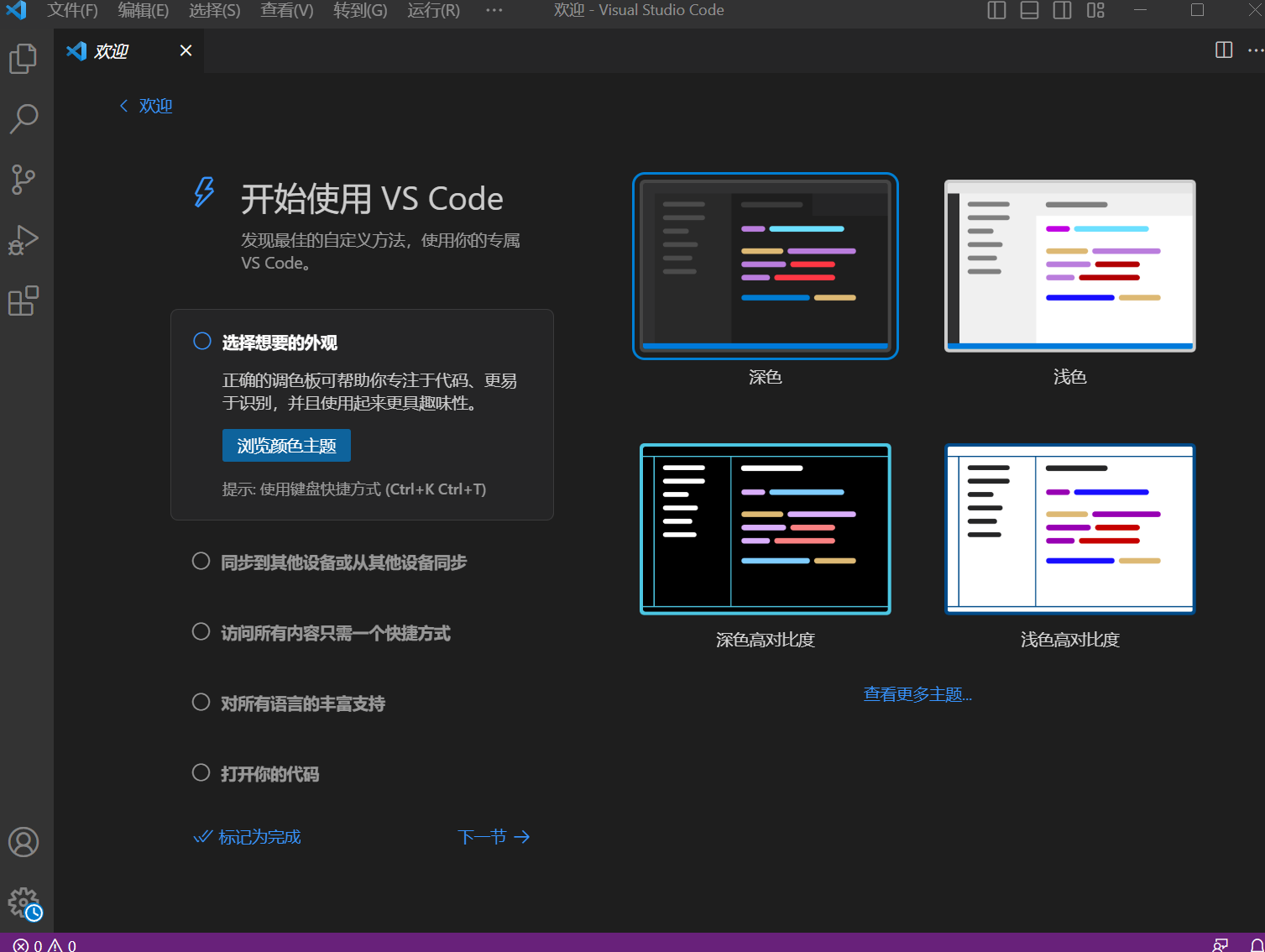
$ pacman -S mingw-w64-x86\_64-gcc --disable-download-timeout

$ pacman -S mingw-w64-x86\_64-make --disable-download-timeout

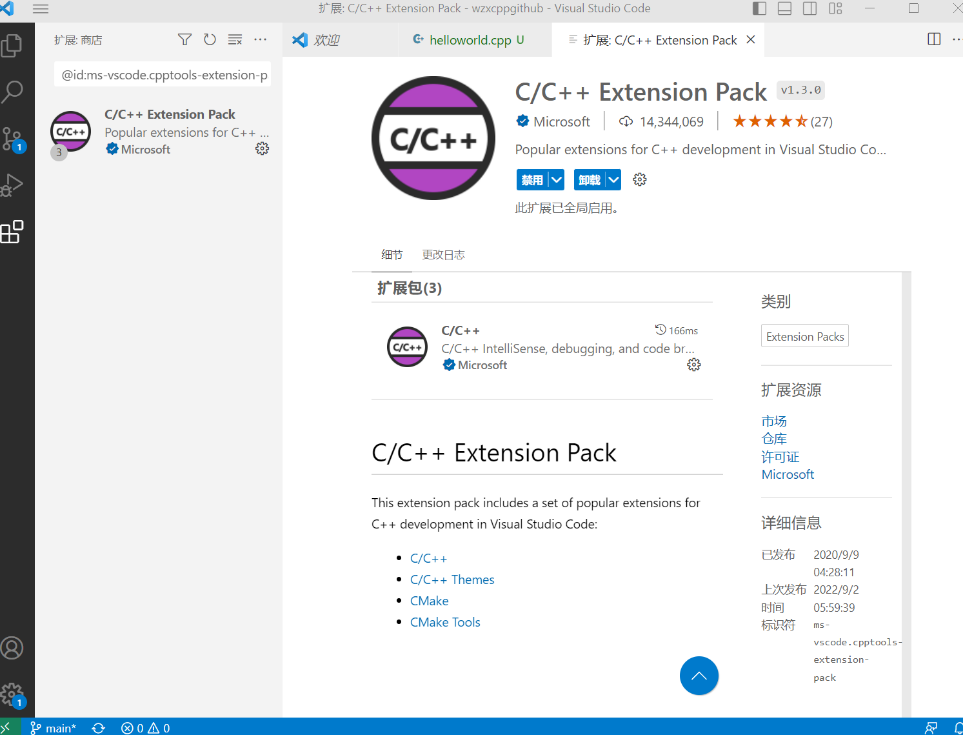
$ pacman -S mingw-w64-x86\_64-gdb --disable-download-timeout

之后等待安装，打开D:\msys64\mingw64\bin查看其中的内容，如果有，表示安装成功。

1. vscode安装

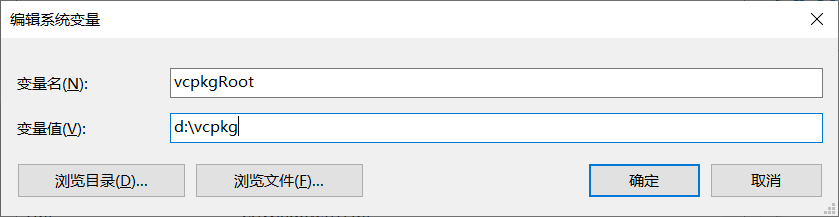
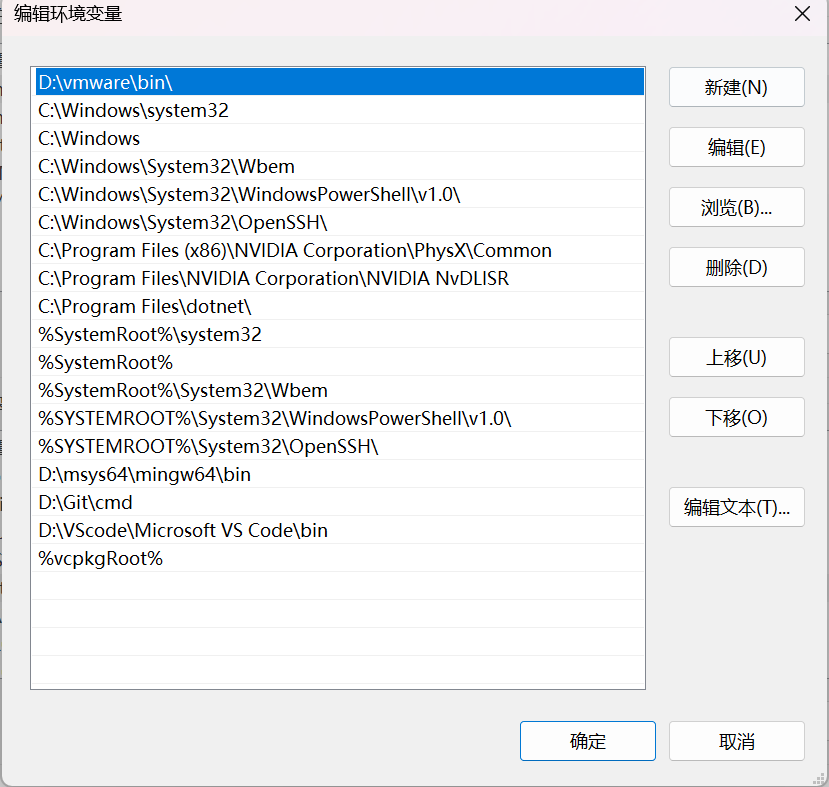


1. 安装中文环境（第一次打开后，系统一般会提示，安装扩展即可），可以根据自己喜好调节背景颜色。
2. 我们新建文件，文件夹选择之前创建好的本地仓库（vscode会自动检测到与云端的链接，之后可以直接在vscode中直接提交云端）。
3. 安装c/c++扩展



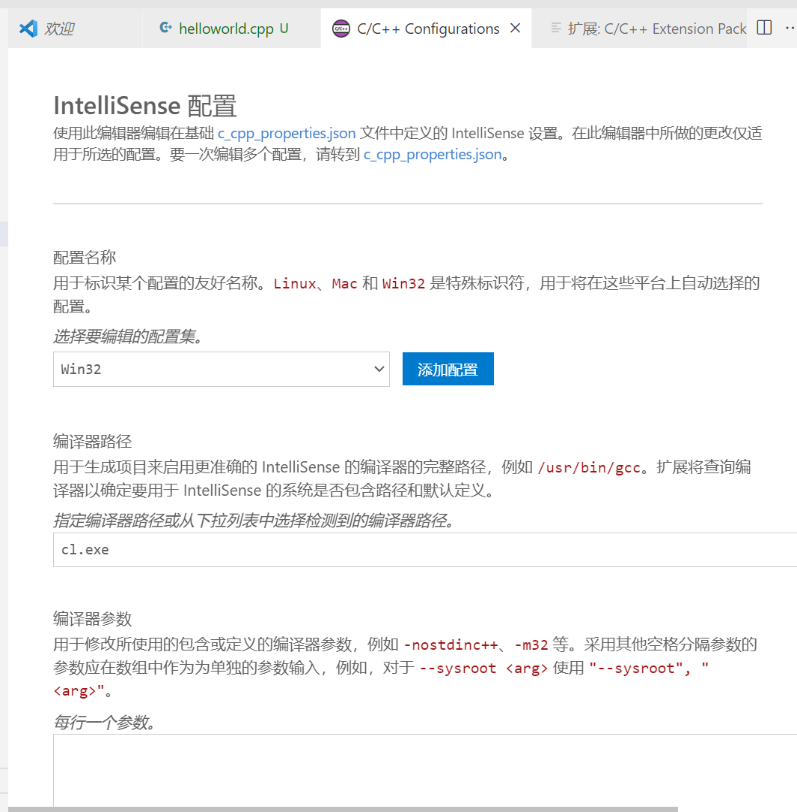
1. 应用vcpkg

重新为vcpkg创建一个系统变量，之后新建一个环境变量。



1. 配置c/c++编译器

按快捷键Ctrl+Shift+P调出命令面板，输入C/C++，选择“Edit Configurations(UI)”进入配置。



之后，分别选择配置选项D:\msys64\mingw64\bin\g++.exe

编译器参数：

-std=c++20

-Mmodules

windows-gcc-x64

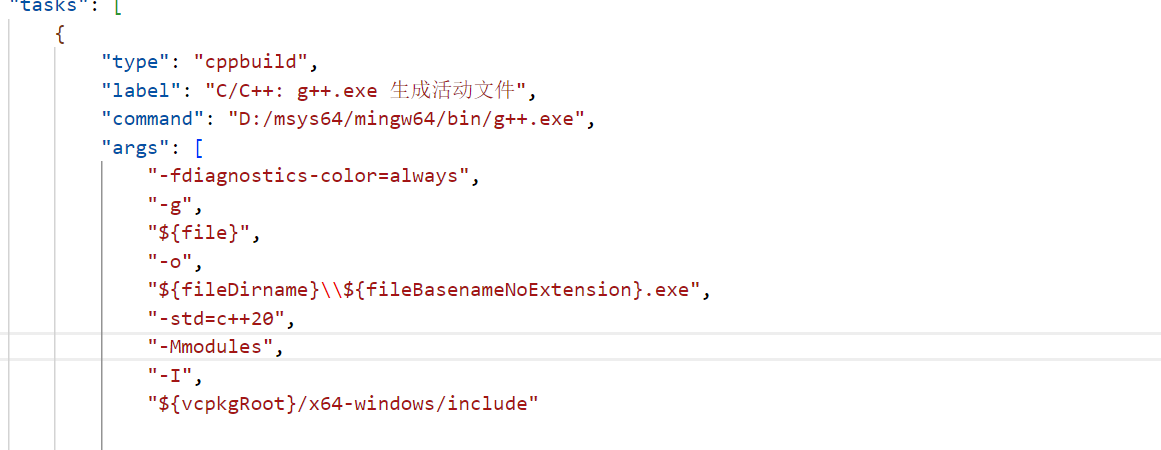
包含路径

${workspaceFolder}/\*\*

${vcpkgRoot}/\*\*

c17，c++20

接下来，创建一个tasks.json文件来告诉vscode如何创建编译程序，按快捷键Ctrl+Shift+P调出命令面板，输入tasks，选择“Tasks:Configure Default Build Task”，选择“C/C++: g++.exe生成活动文件”



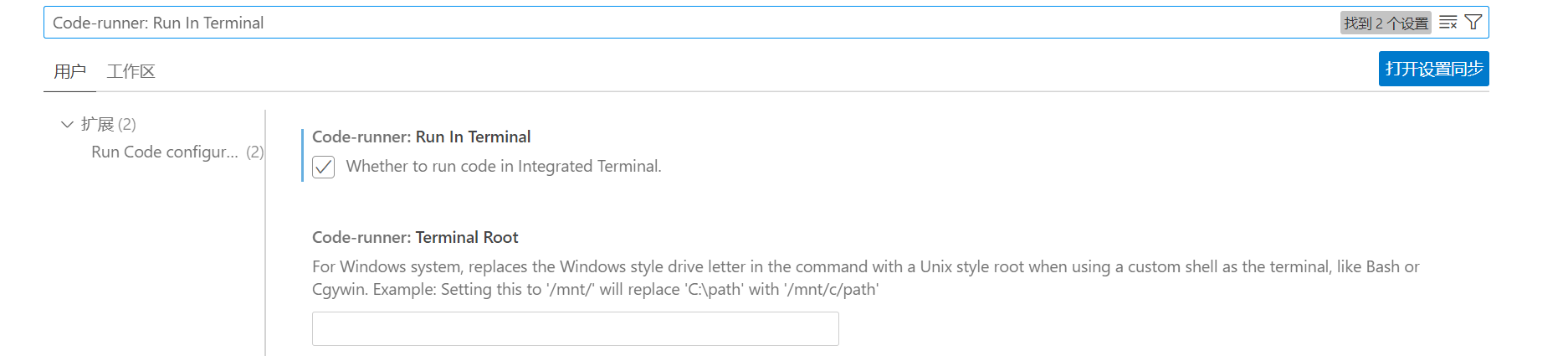
在图片位置，添加图片最下方四行，之后我们可以创建helloworld代码，按ctrl+f5，进行调试，看是否能够运行，还以实验一下vcpkg是否可以使用。

1. code runner使用

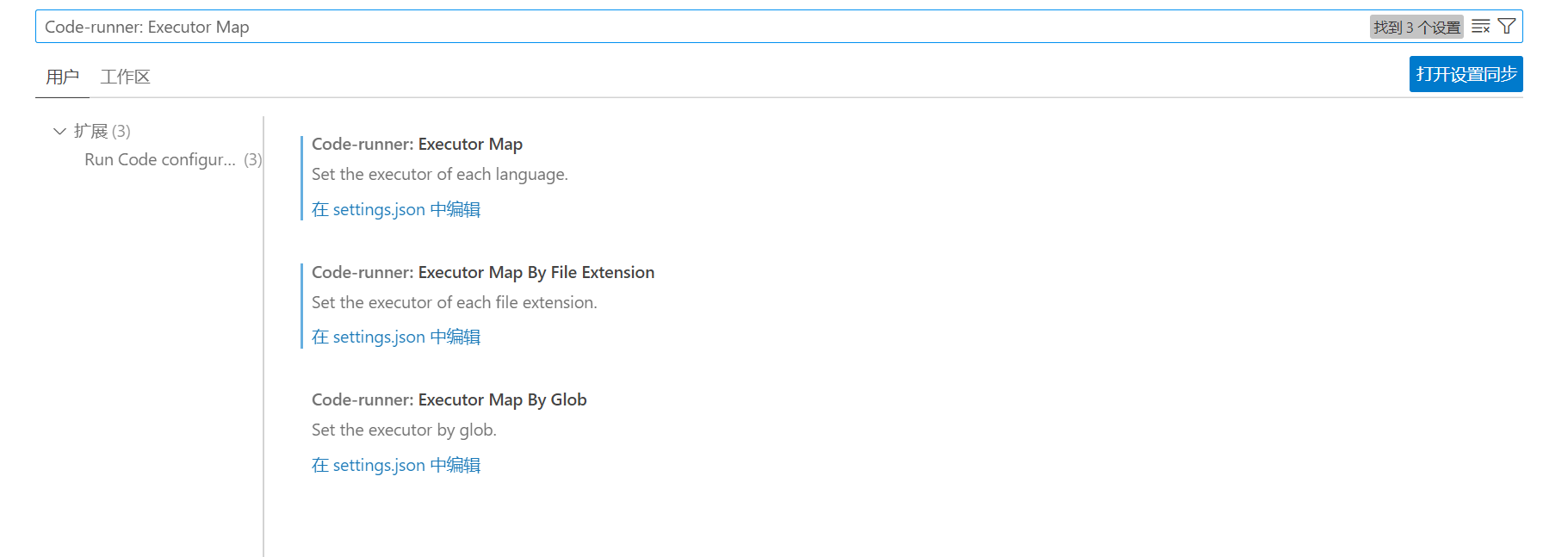
虽然，f5也可以用来调试，但是调试信息都会显示，显得过于冗杂，我们可以在扩展中安装code runner，方便我们进项调试运行



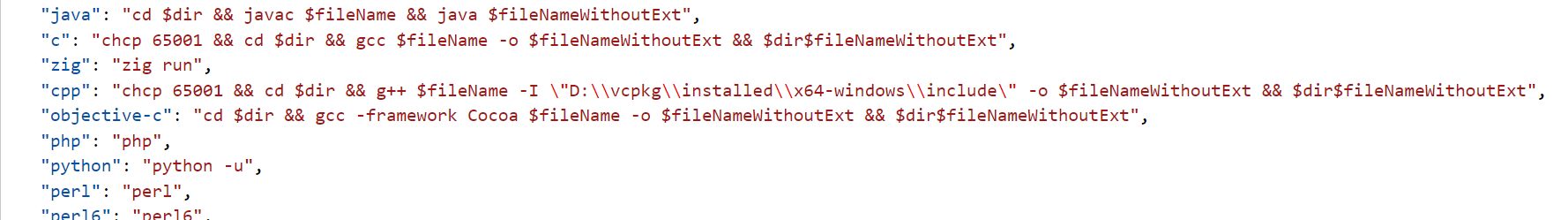
之后，我们打开设置，输入Code-runner: Run In Terminal，勾选Whether to run code in Integrated Terminal，以后我们便可以使用code runner在终端模式下调试运行程序



打开设置，查找Code-runner: Executor Map



在setting.json中编辑



此时code runner及支持c++20与vcpkg

至此，c++环境就完全配置成功了，这个过程看似十分麻烦，但实则上，这是也是一次全新的体验与尝试。在整个过程之中，会不断的遇到各种各样的问题，但这也是走上计算机这一人生道路所必须经历的。在一遍遍的安装，出现问题，重新安装，以及查询资料之中，渐渐的也对这一操作渐渐熟悉了起来，不再是一开始的感到陌生，以及无从下手。这次实验，是我开始接触一个全新领域的起点，让我开始敢于接触一些看似十分麻烦的东西，增强了我的动手实践能力。

与此同时，我也明白，这仅仅是一个开始，无论是c++这门课程的开始，还是我未来人生道路的开始。未来的如何去走，还需要我继续的努力与奋斗!