

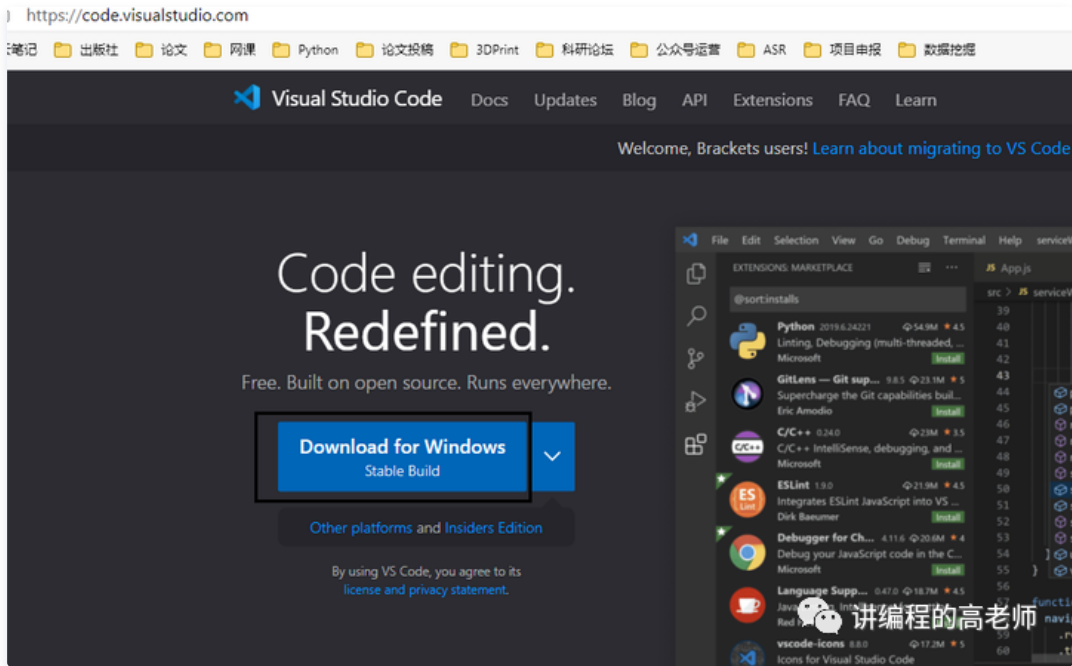
VS Code配置C语言开发环境的超详细教程

2021-03-18 阅读 1.9K

今年有个C语言的课程，以往上C语言课都是用VC++6 作为开发环境，VC6实在是太古老了，这个学期想用Visual Studio Code来配置C语言的开发环境。把整个环境变量配置、工具安装使用的步骤总结如下，因为是针对初学者，所以步骤非常详细。

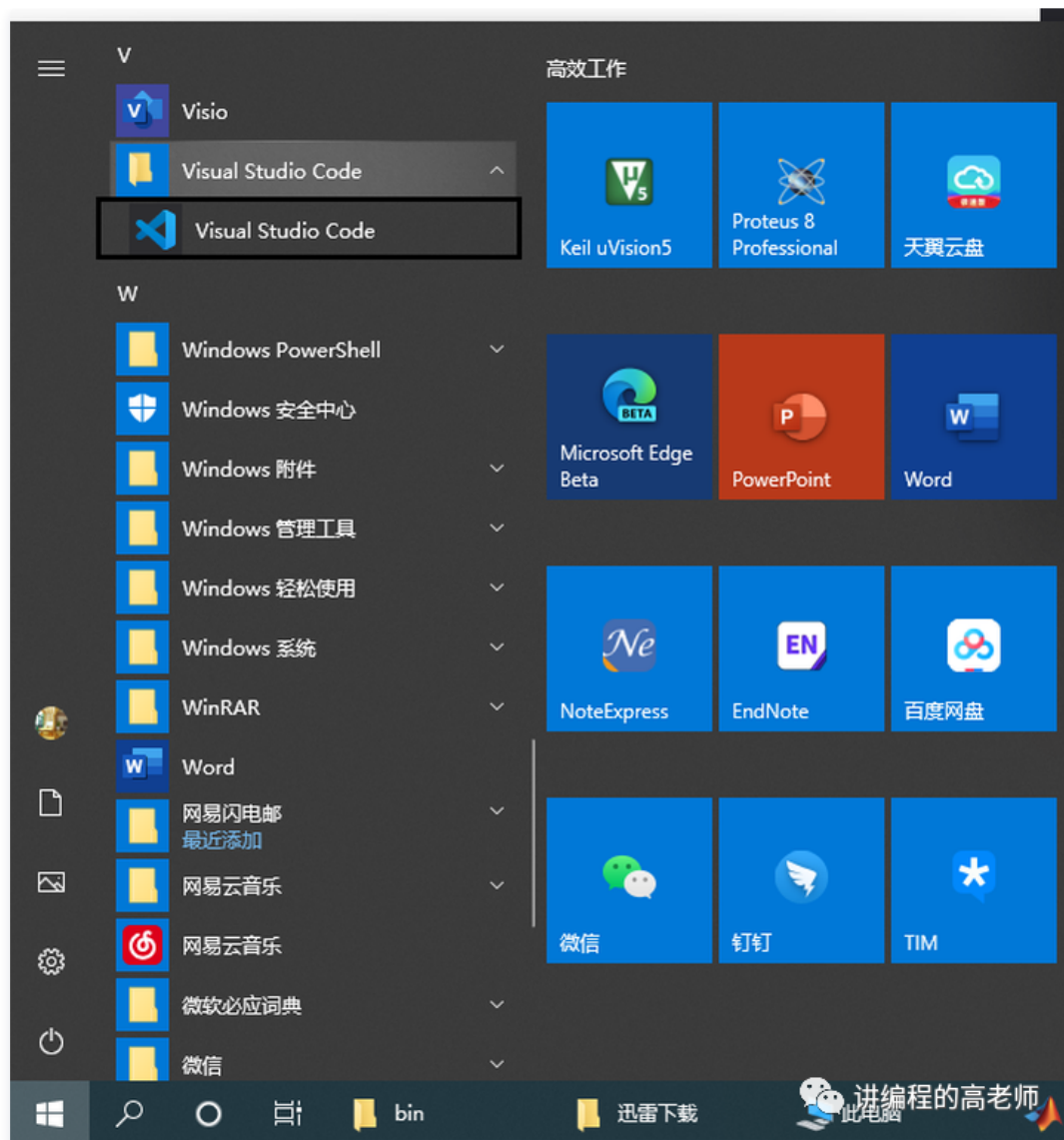
安装并配置Visual Studio Code

下载VSCode，下载地址为：<https://code.visualstudio.com/>，在网页中点击"Download for Windows"按钮即可下载。当然，还可以点击按钮右侧紧挨的向下的箭头选择不同操作系统的VSCode版本。

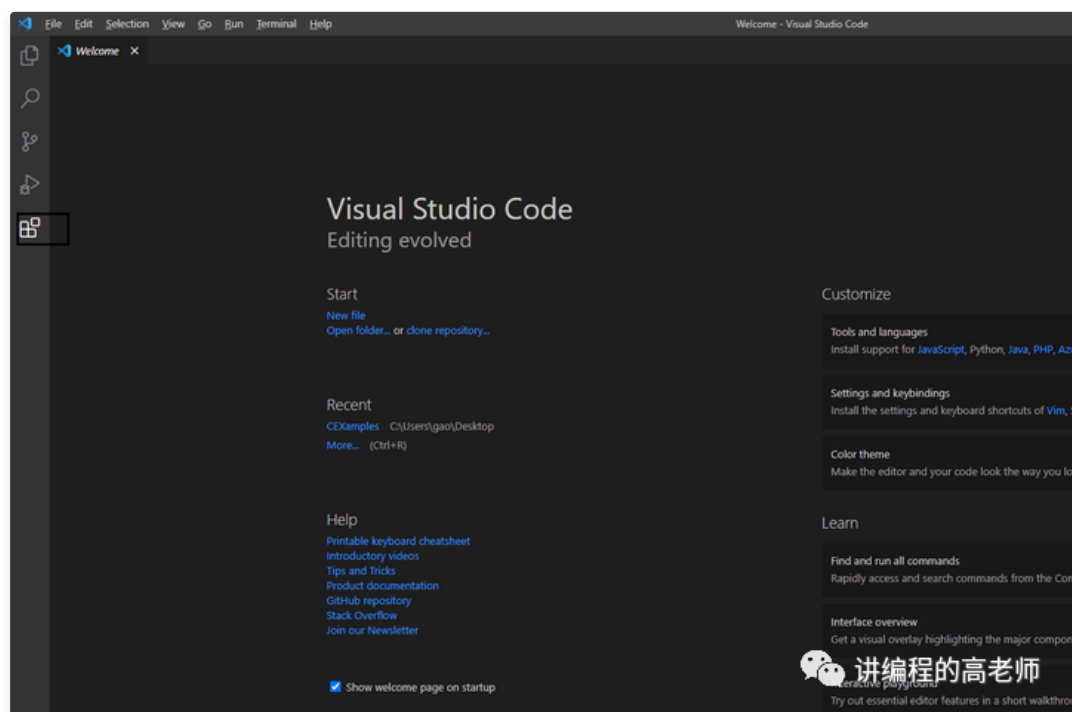


下载下来是一个exe的安装文件，"VSCodeUserSetup-x64-1.53.2.exe"。如果你觉得下载速度慢，可以复制下载链接然后使用迅雷等下载工具下载。

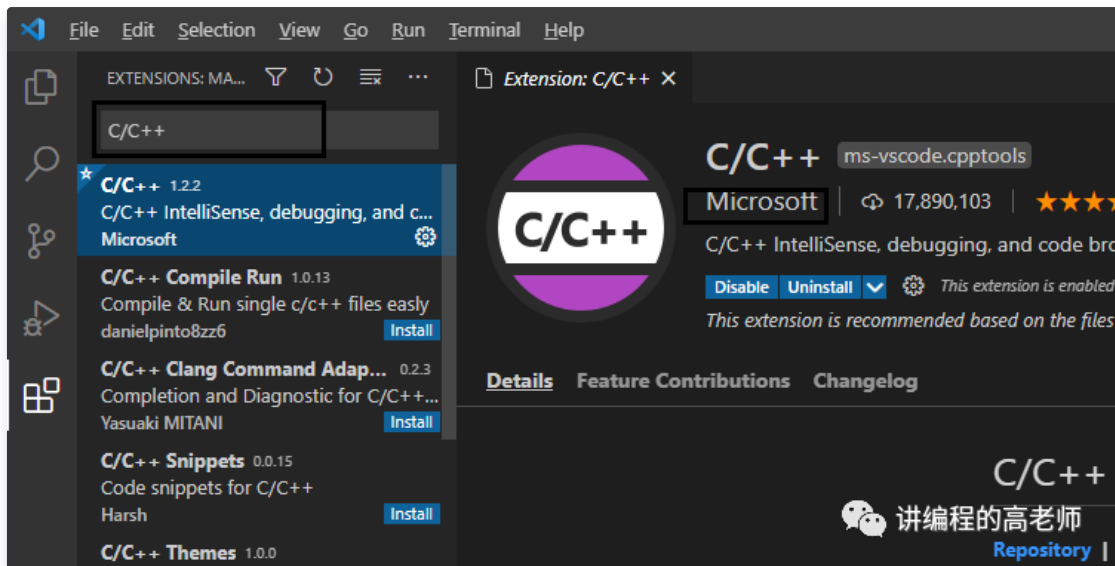
VSCode的安装比较简单，根据提示一路傻瓜式安装，完成后在开始菜单下就会有"Visual Studio Code"了，点击打开。



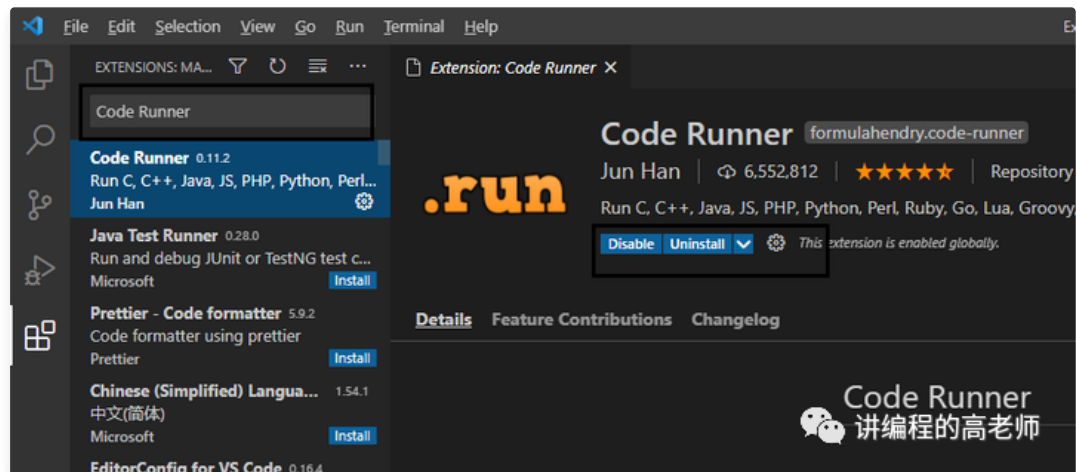
打开后，如下图。左侧边栏有几个快速的按钮，点击最下面的“Extensions”（扩展）按钮。



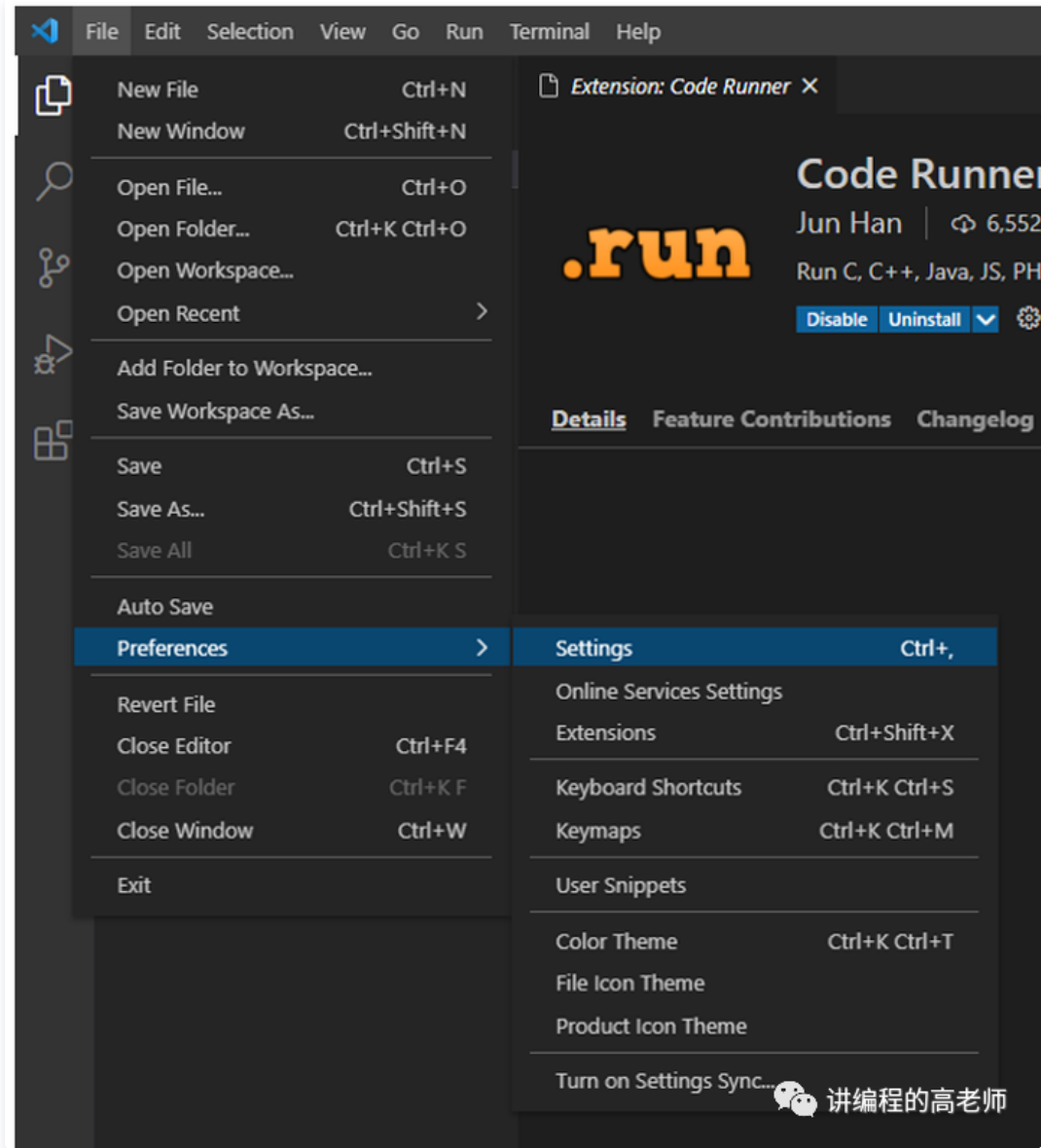
在随后出现的“扩展市场”的搜索框中输入“C/C++”，在随后出现的列表中选择对应的扩展，确认是Microsoft家的，就点击“Install”按钮即可安装。



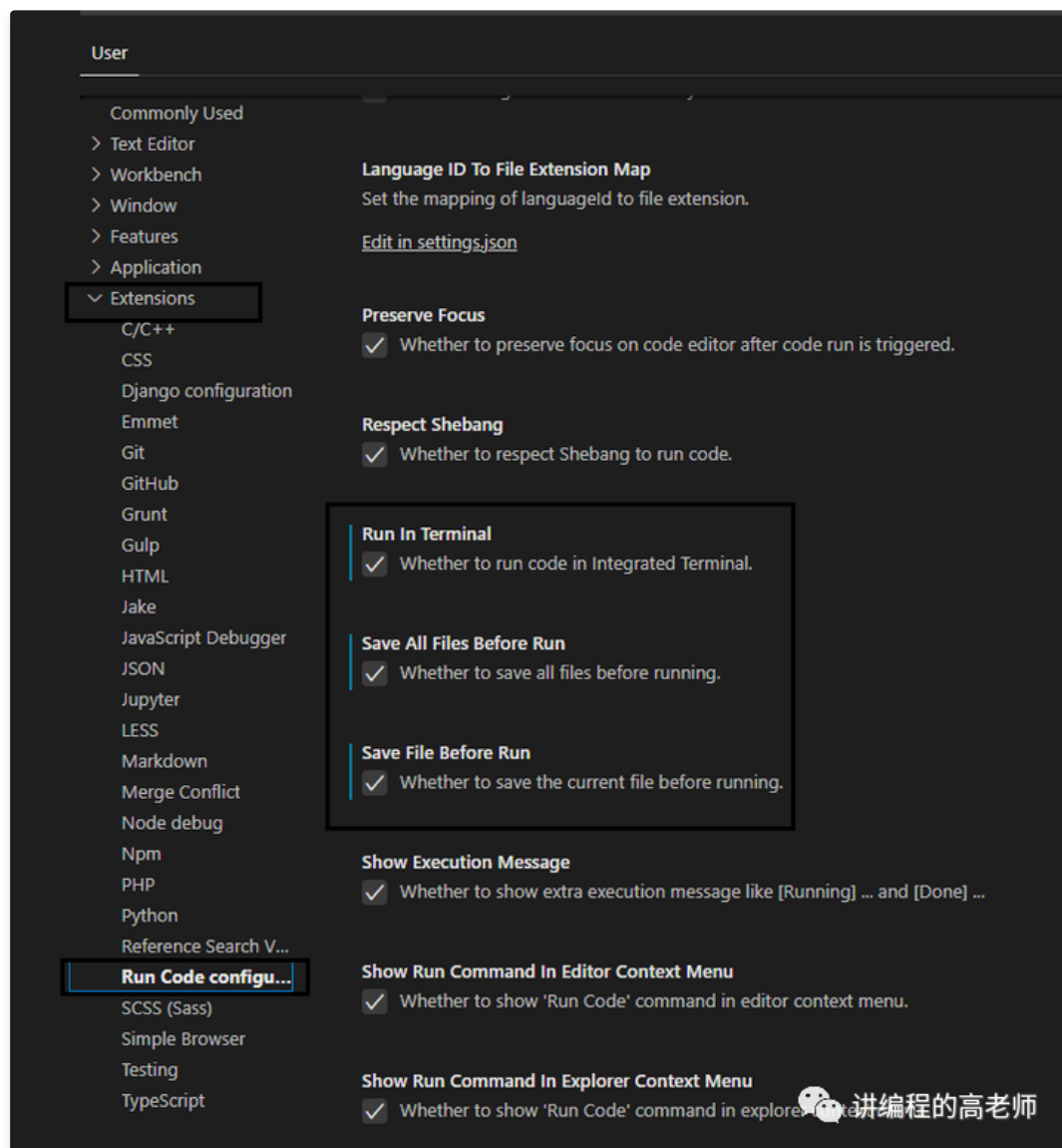
安装完“C/C++”扩展后，安装“Code Runner”扩展，方法相同。



俩扩展安装完成后，是对“Code Runner”扩展的配置，在“File”菜单下找到“Preference”再找到“Settings”子菜单。



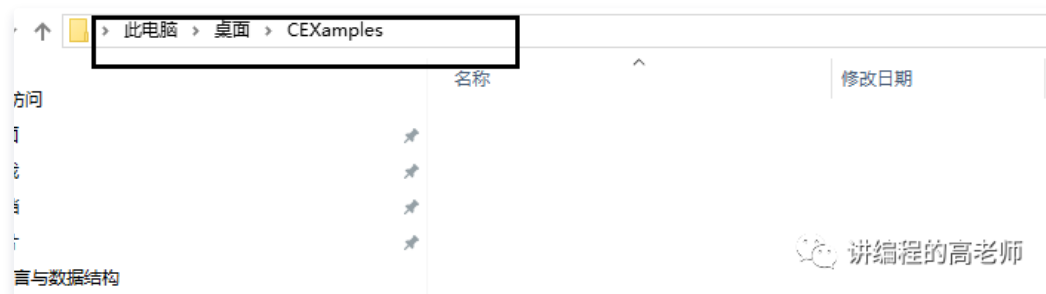
在随后的Settings页面中，将“Run Code Configuration”中的“Run In Terminal”、“Save File Before Run”前面的复选框选中。



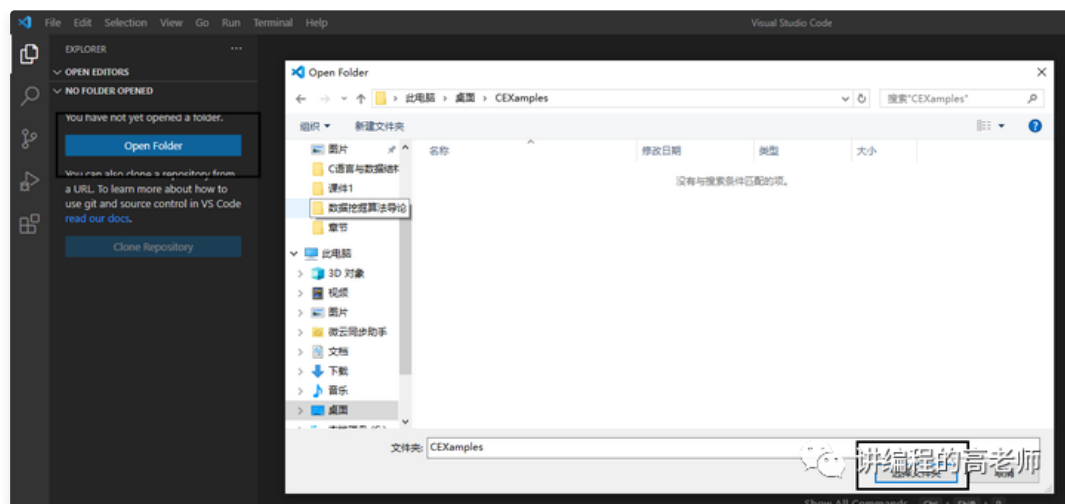
VSCode及其扩展的安装、位置完成。

开发C语言工程

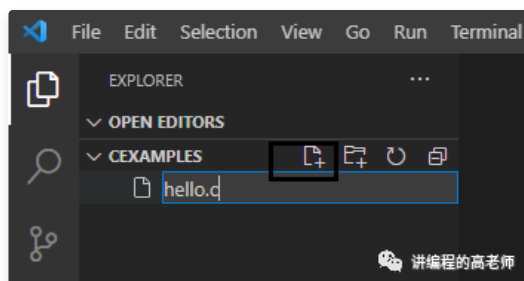
在你喜欢的位置创建一个文件夹，建议使用纯英文字符的名。此处以在桌面上创建“CEExamples”文件夹为例，



随后，打开VSCode，然后在“File”菜单下找到“Open Folder”子菜单，并点击。定位到刚刚创建的新文件夹，然后点击“选择文件夹”按钮。当然，如果你刚打开VSCode，没有选定文件夹，它也会出现一个“Open Folder”按钮，你通过这个按钮选择刚刚创建的文件夹效果是一样的。

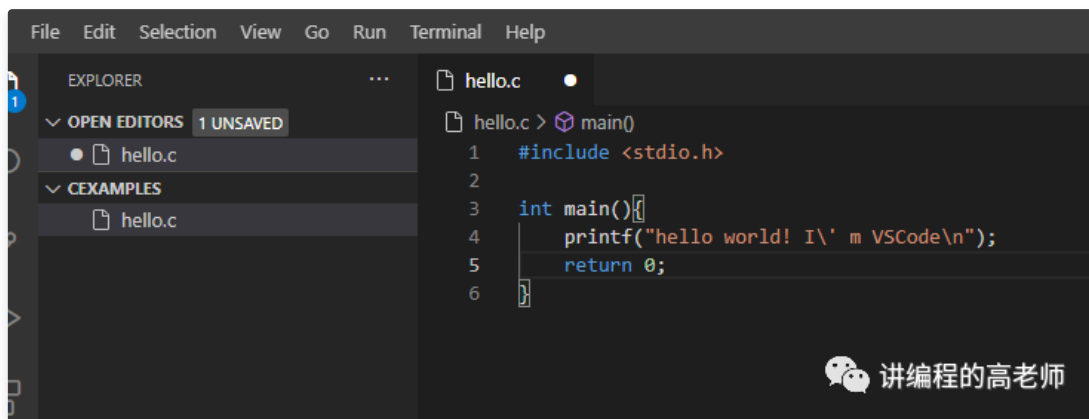


鼠标移动到打开的文件夹处会出现4个快捷按钮，点击左边第一个就可以给文件夹创建一个新的文件，命名为"hello.c"。



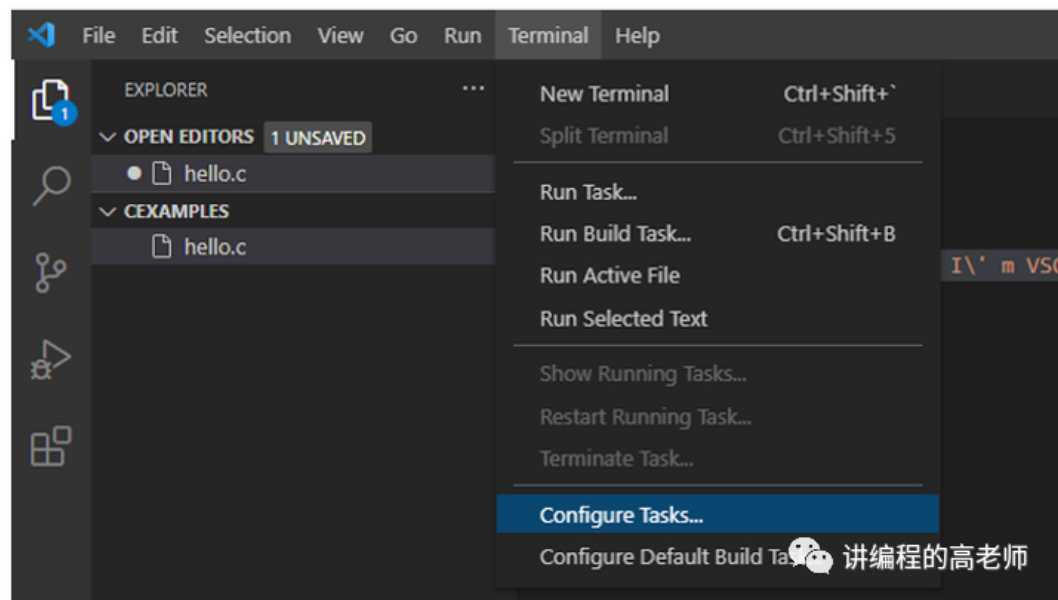
在"hello.c"文件中输入简单的命令。

```
#include <stdio.h>int main(){
    printf("hello world! I\' m VSCode\n");
    return 0;}
```

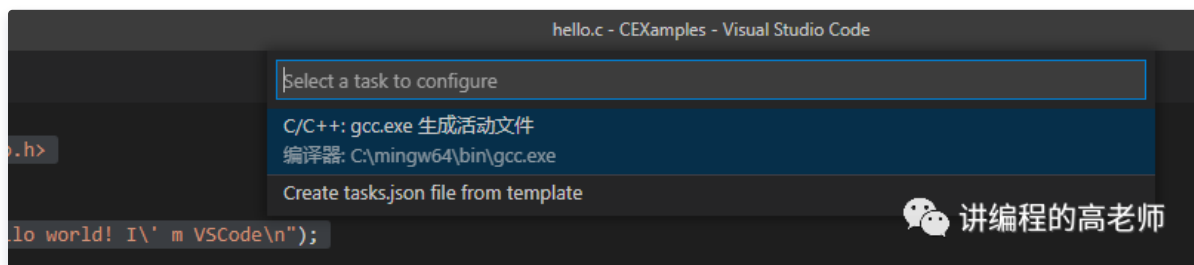


重点来了，代码编辑完成后还不能直接运行。还需要配置俩json文件。

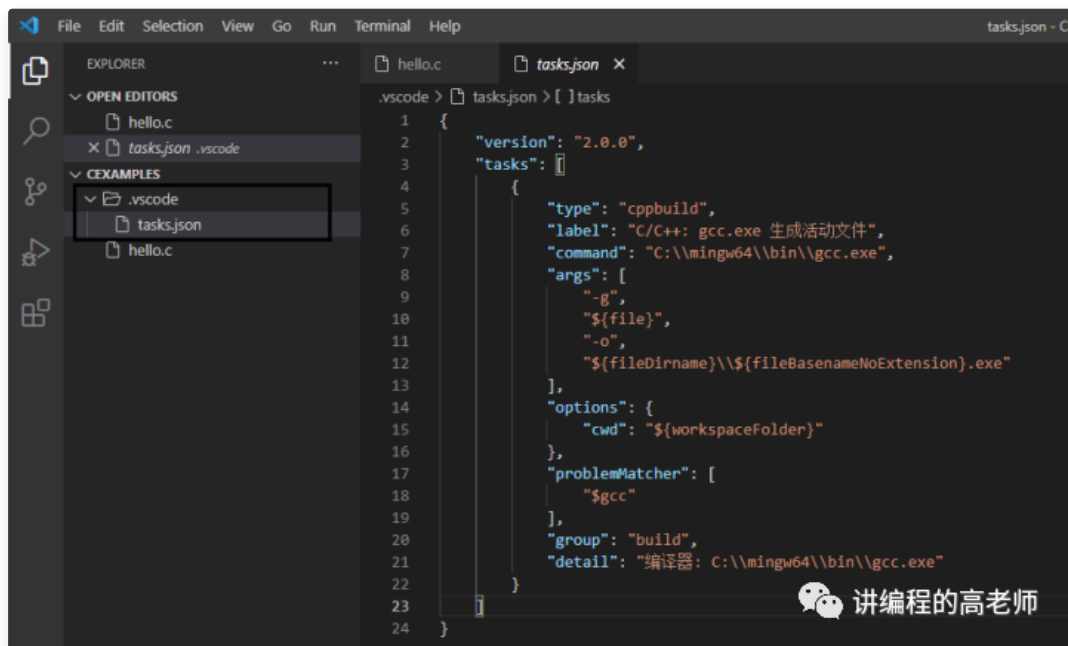
首先，在"Terminal"菜单下点击"Configure Tasks..."子菜单。



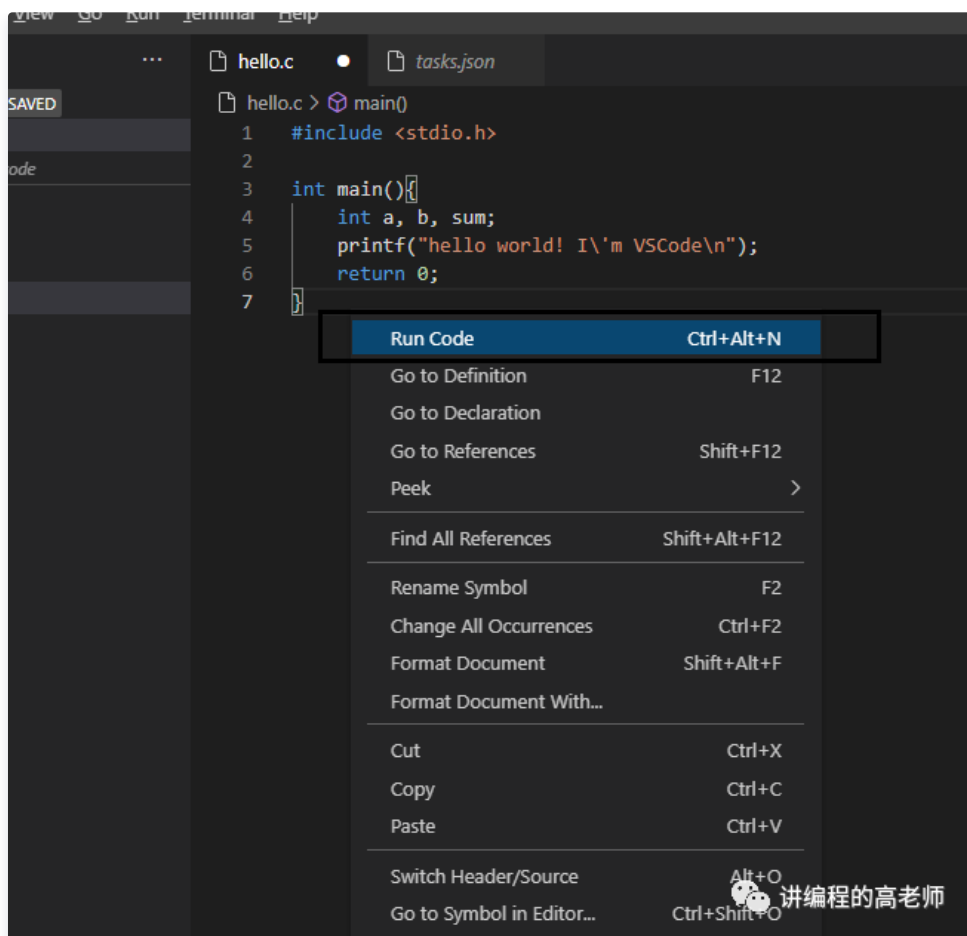
在随后出现的弹出框中选择“C/C++:gcc.exe生成活动文件”



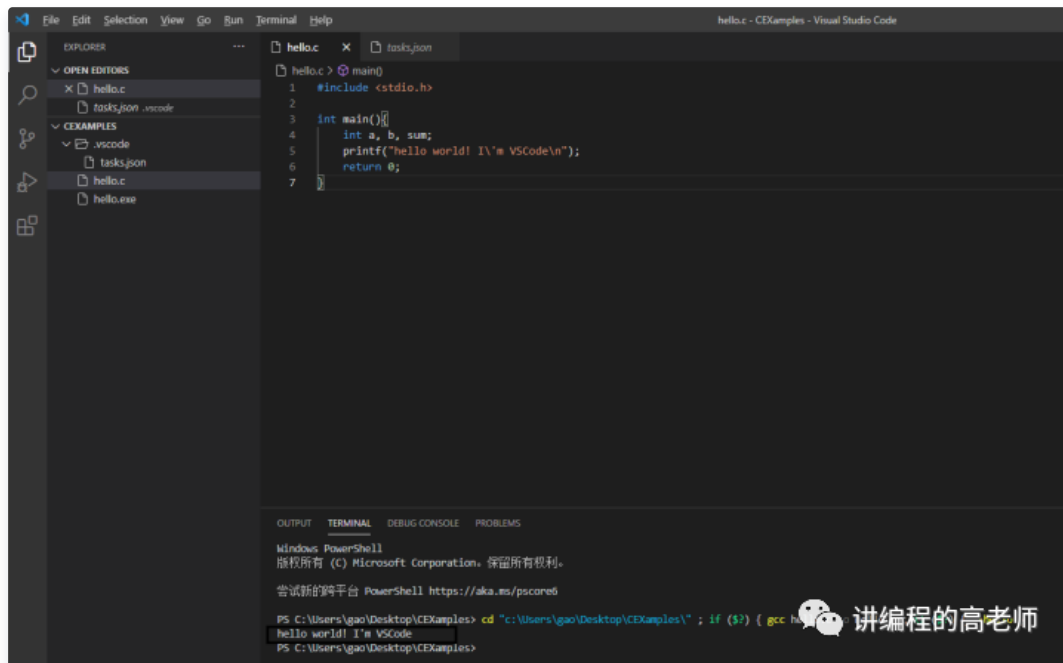
随后，在“CEXAMPLES”文件夹中会自动生成一个“.vscode”的子文件夹，在这个子文件夹下自动创建一个“tasks.json”的文件。不用做任何改动。



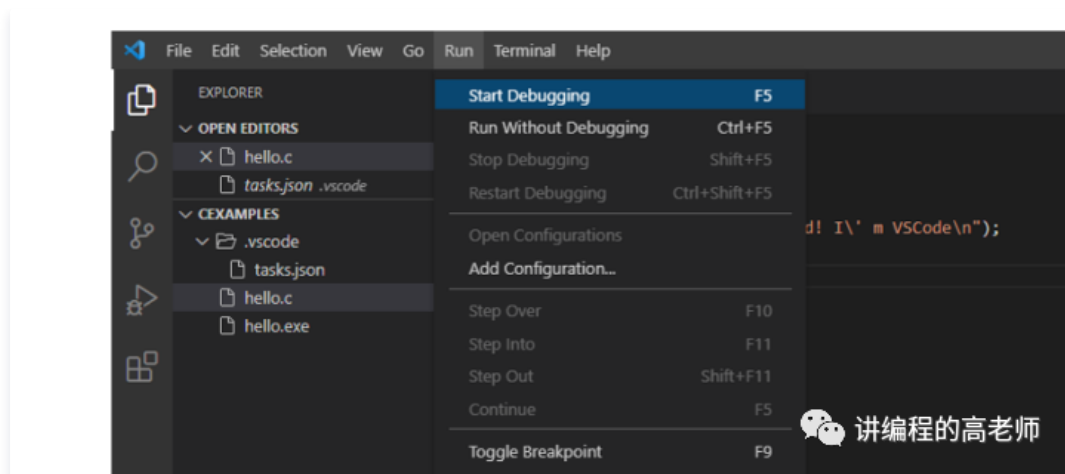
最后，回到“hello.c”文件，单击鼠标右键选择“Run Code”即可运行代码。



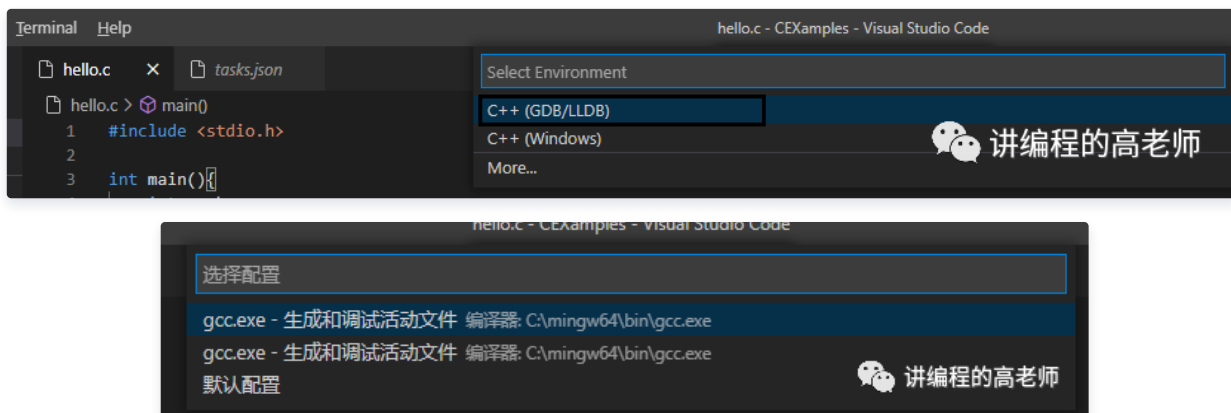
运行结果会在下面出现的“Terminal”中显示



如果想要调试代码的话呢？点击“Run”菜单下的“Start Debugging”子菜单。

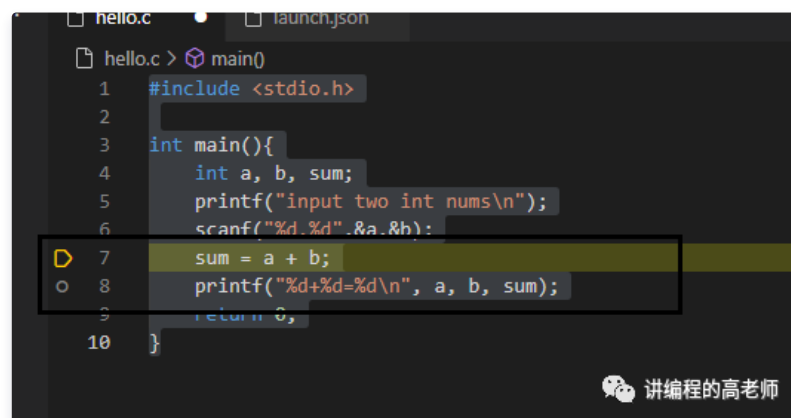


在随后出现的弹出框中选择“C++(GDB/LLDB)”，再选择“gcc.exe”。

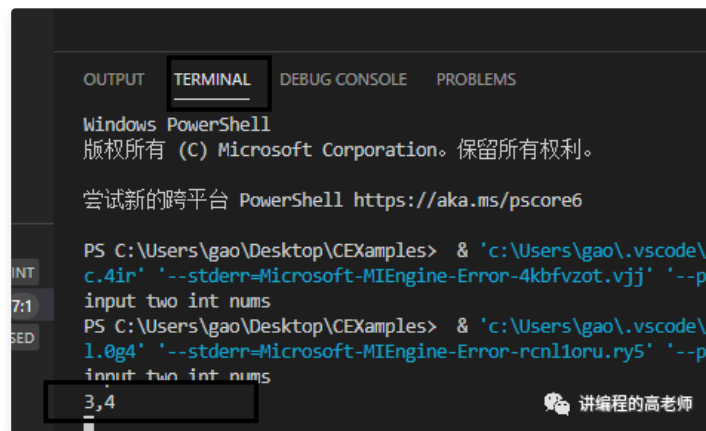


这个时候就可以进入调试，如果我们将代码改一下，并设置俩断点。

```
#include <stdio.h>int main(){
    int a, b, sum;
    printf("input two int nums\n");
    scanf("%d,%d",&a,&b);
    sum = a + b;
    printf("%d+%d=%d\n", a, b, sum);
    return 0;}
```



这个时候，进入调试如下：运行到scanf那一行后在terminal中等待我们输入，假设输入“3,4”，然后回车。

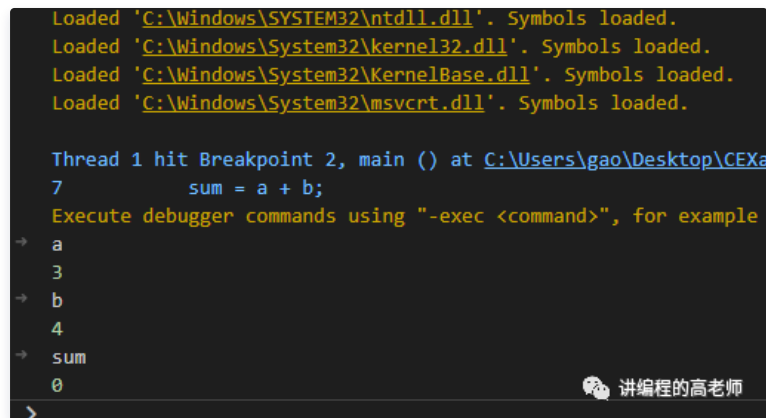


```
Windows PowerShell
版权所有 (C) Microsoft Corporation。保留所有权利。

尝试新的跨平台 PowerShell https://aka.ms/pscore6

PS C:\Users\gao\Desktop\CEXamples> & 'c:\Users\gao\.vscode\exte
c.4ir' '--stderr=Microsoft-MIEngine-Error-4kbfvzot.vjj' '--pi
input two int nums
PS C:\Users\gao\Desktop\CEXamples> & 'c:\Users\gao\.vscode\exte
1.0g4' '--stderr=Microsoft-MIEngine-Error-rcn1loru.ry5' '--pi
input two int nums
3,4
```

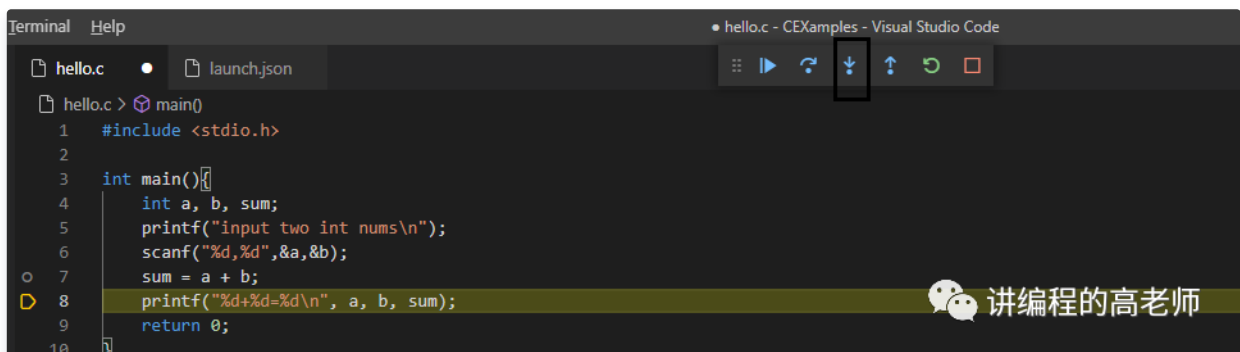
这个时候，程序运行到第7行会暂停，如果在下面的“DEBUG CONSOLE”中输入变量名，调试窗口会返回我们变量在程序运行到当前调试状态下的值。



```
Loaded 'C:\Windows\SYSTEM32\ntdll.dll'. Symbols loaded.
Loaded 'C:\Windows\System32\kernel32.dll'. Symbols loaded.
Loaded 'C:\Windows\System32\KernelBase.dll'. Symbols loaded.
Loaded 'C:\Windows\System32\msvcrt.dll'. Symbols loaded.

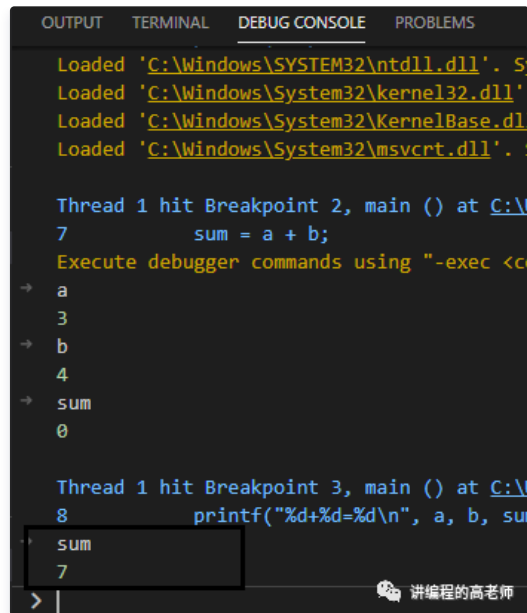
Thread 1 hit Breakpoint 2, main () at C:\Users\gao\Desktop\CEXa
7      sum = a + b;
Execute debugger commands using "-exec <command>", for example
→ a
3
→ b
4
→ sum
0
```

点击上方的“Step Into”图标，会运行到下一行。



```
hello.c • launch.json
hello.c > main()
1  #include <stdio.h>
2
3  int main(){
4      int a, b, sum;
5      printf("input two int nums\n");
6      scanf("%d,%d",&a,&b);
7      sum = a + b;
8      printf("%d+%d=%d\n", a, b, sum);
9      return 0;
10 }
```

这个时候再在“DEBUG CONSOLE”中输入sum，这个变量的值已经变成a+b的值7了。



```
OUTPUT  TERMINAL  DEBUG CONSOLE  PROBLEMS

Loaded 'C:\Windows\SYSTEM32\ntdll.dll'. Sy
Loaded 'C:\Windows\System32\kernel32.dll'
Loaded 'C:\Windows\System32\KernelBase.dll'
Loaded 'C:\Windows\System32\msvcrt.dll'. S

Thread 1 hit Breakpoint 2, main () at C:\V
7      sum = a + b;
Execute debugger commands using "-exec <co

→ a
3
→ b
4
→ sum
0

Thread 1 hit Breakpoint 3, main () at C:\V
8      printf("%d+%d=%d\n", a, b, sum

sum
7

> |
```

小结

好了，使用VSCode开发C语言的配置搞定了，还是比较简单方便的。而且VSCode的编程界面比VC++6好太多了，它的代码自动提示什么的也比VC++6强大。