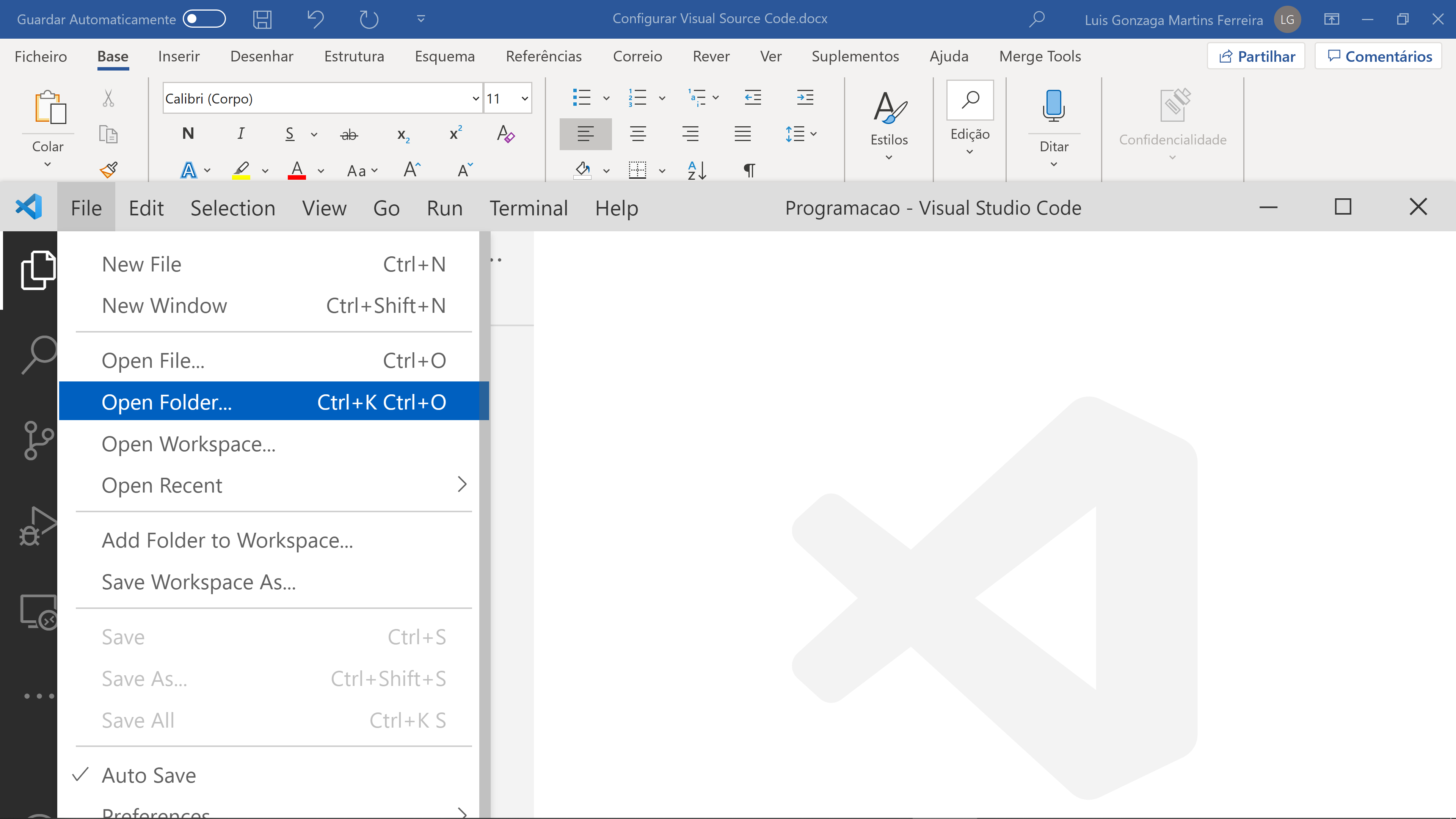
# Visual Studio Code (VSC)

## Preparação de Plataforma para trabalhar em C

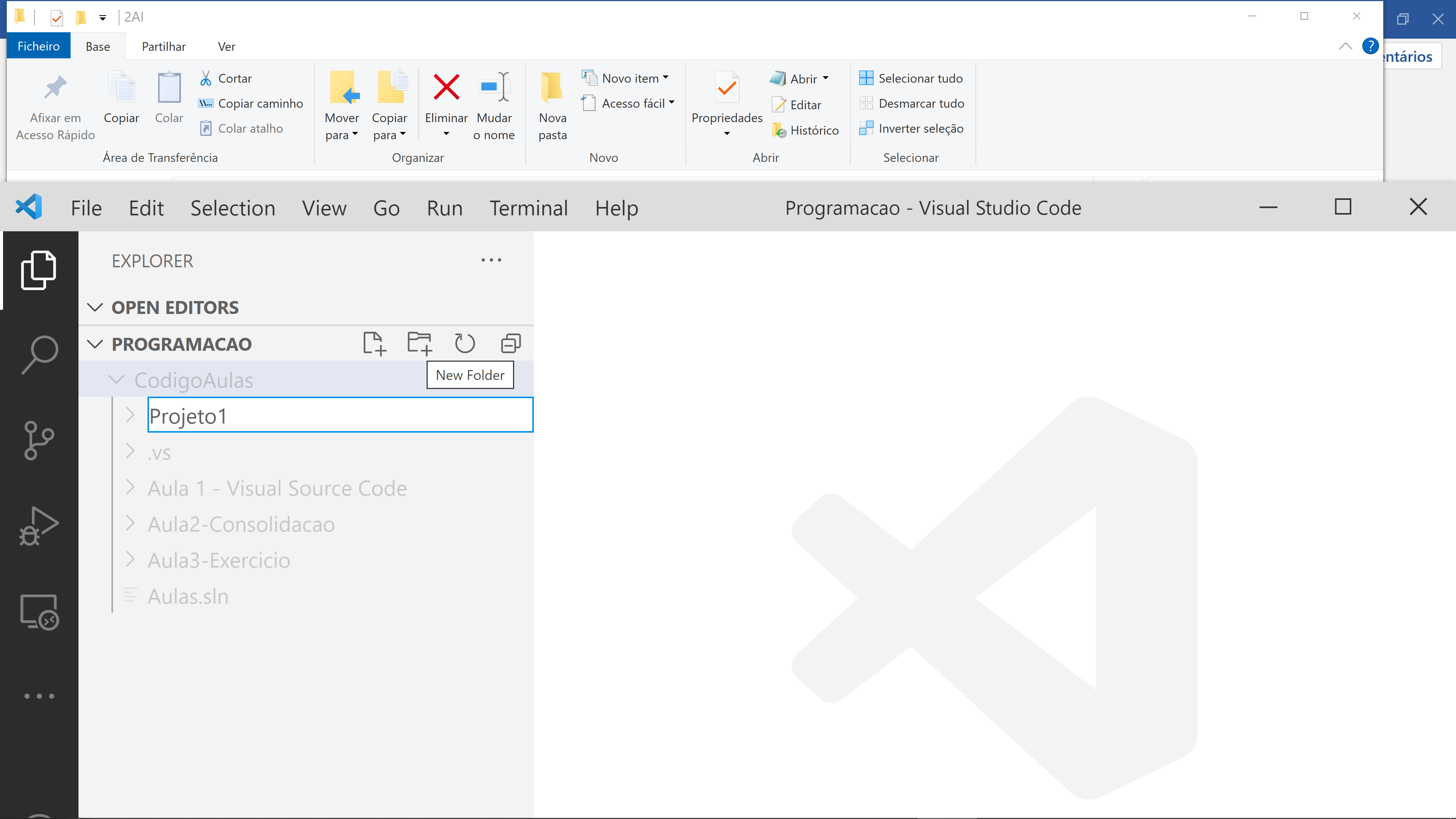
1. Criar pasta (Folder) no disco para armazenar o código produzido

Sugestão: c:\temp\Programao

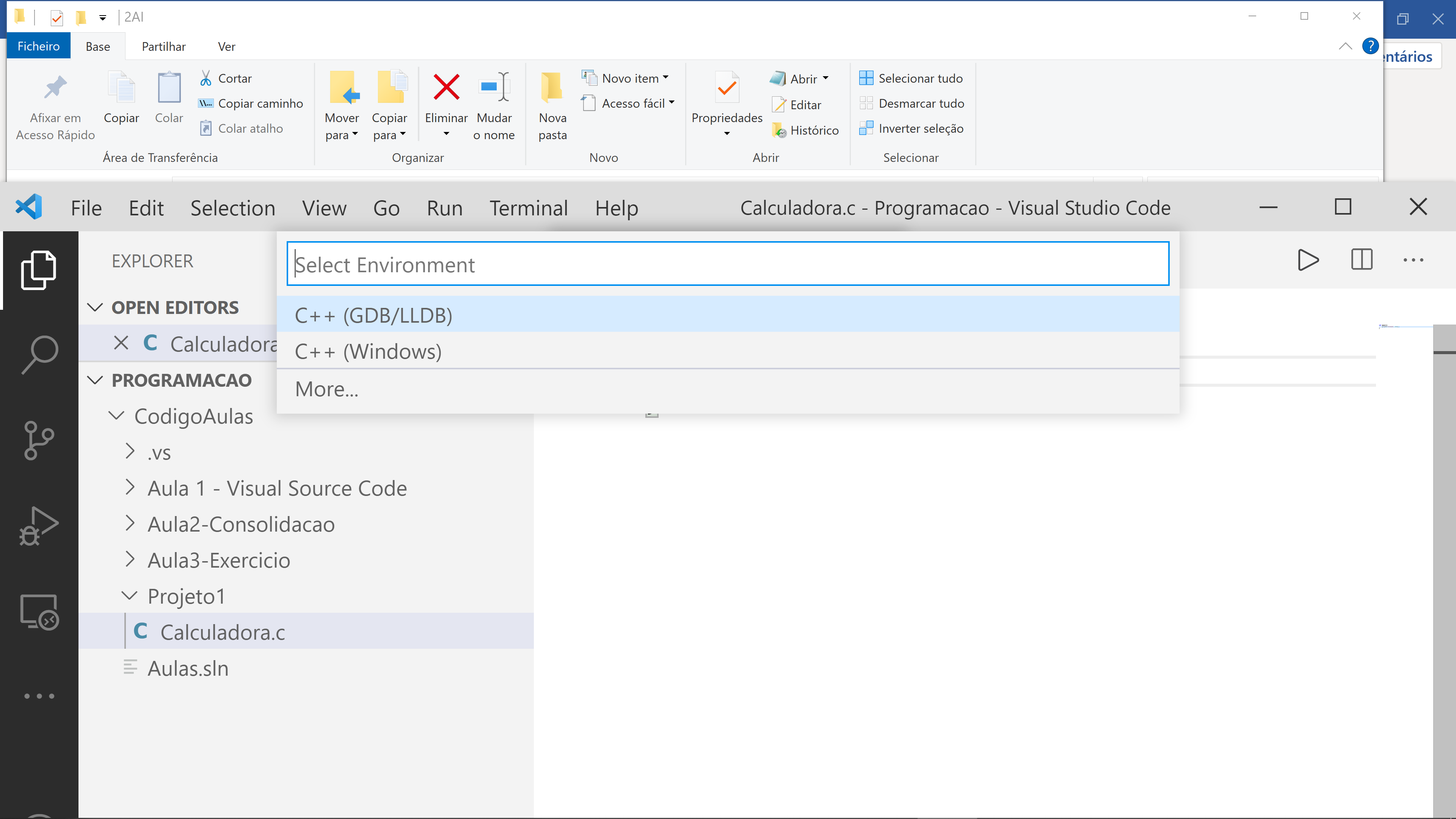
1. Iniciar VSC e colocar-se (*Open Folder*) na pasta que criou (c:\temp\Programao)



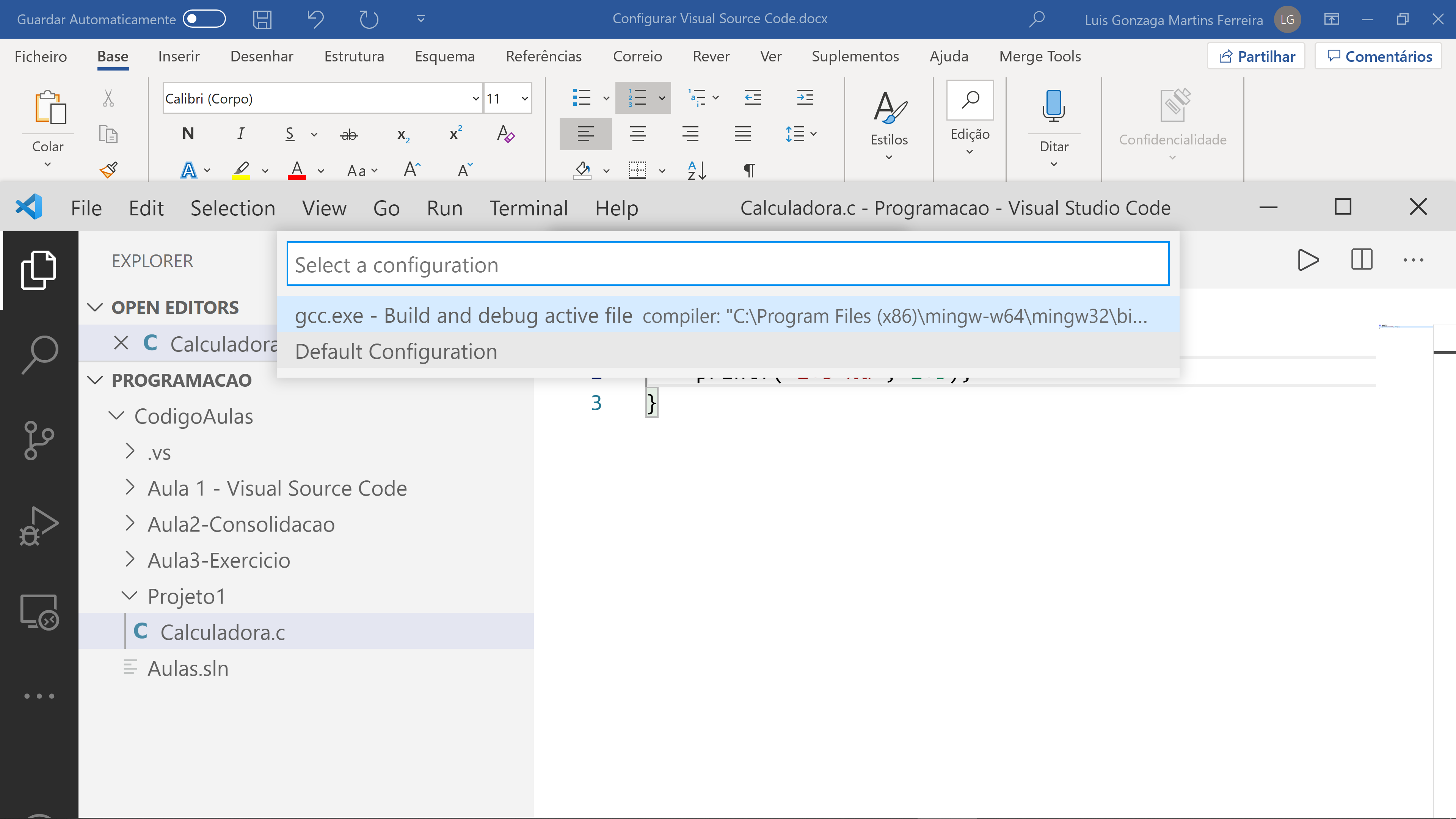
1. Crie uma nova pasta com o nome Projeto1



1. Criar ficheiro para código *Calculadora.c*. Documentar!
2. Criar ficheiro para definições de funções *Operacoes.h*. Documentar!
3. Criar ficheiro para implementação das funções *Operacoes.c*. Documentar!
4. Inserir o código necessário em cada um dos ficheiros
5. Compilar e preparar para Debug: **hipótese 1**
   1. *F5* ou *Run🡪Start Debugging*
   2. Escolher C++(GDB/LLDB)

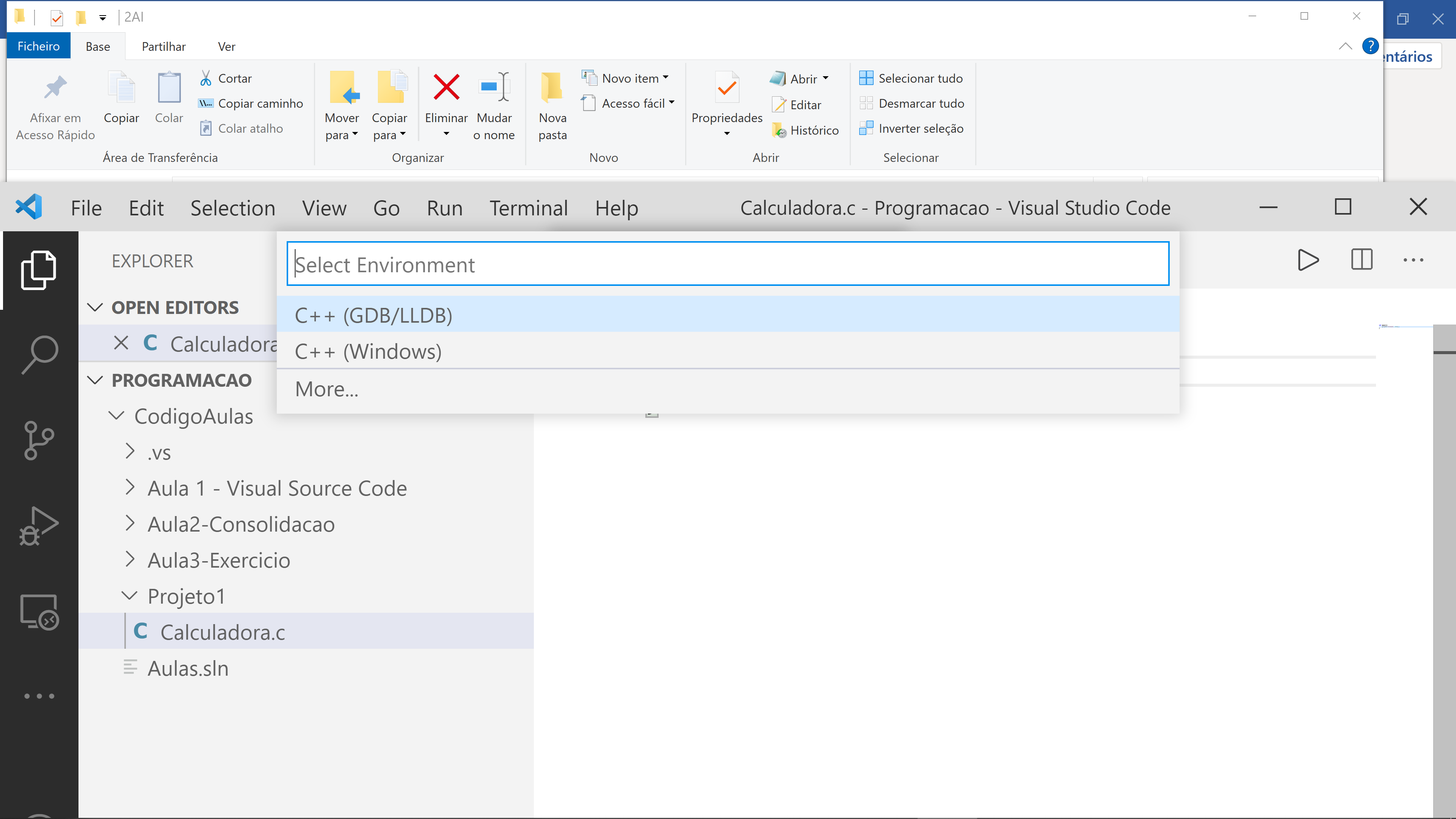


* 1. Escolher “*gcc.exe*”

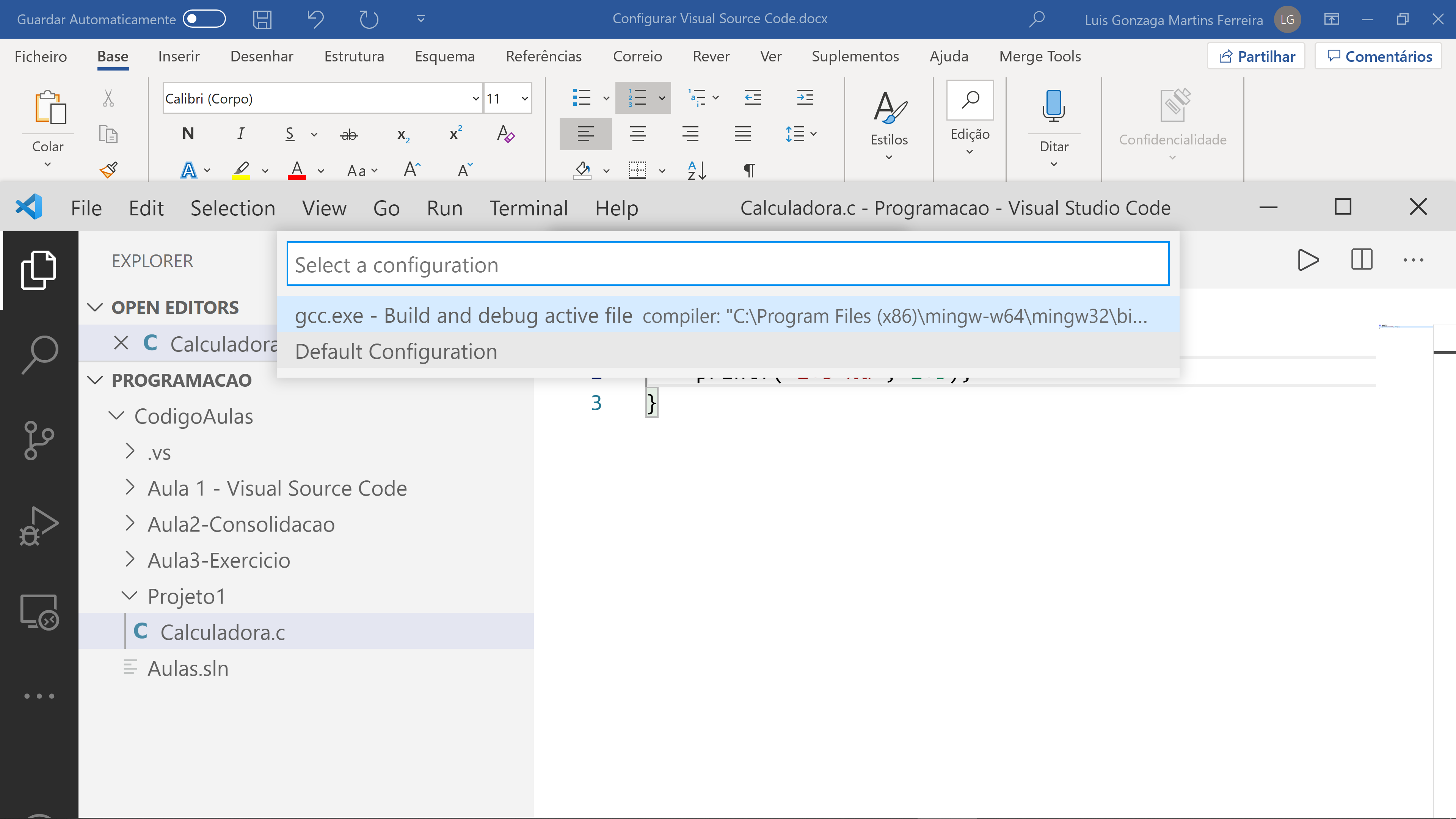


* 1. Explorar o Debugg
     1. Breakpoints
     2. Step into | step over

1. Compilar e preparar para Debug: **hipótese 2**
   1. *F5* ou *Run🡪Start Debugging*
   2. Escolher C++(GDB/LLDB)



* 1. Escolher “*Default Configuraation”*



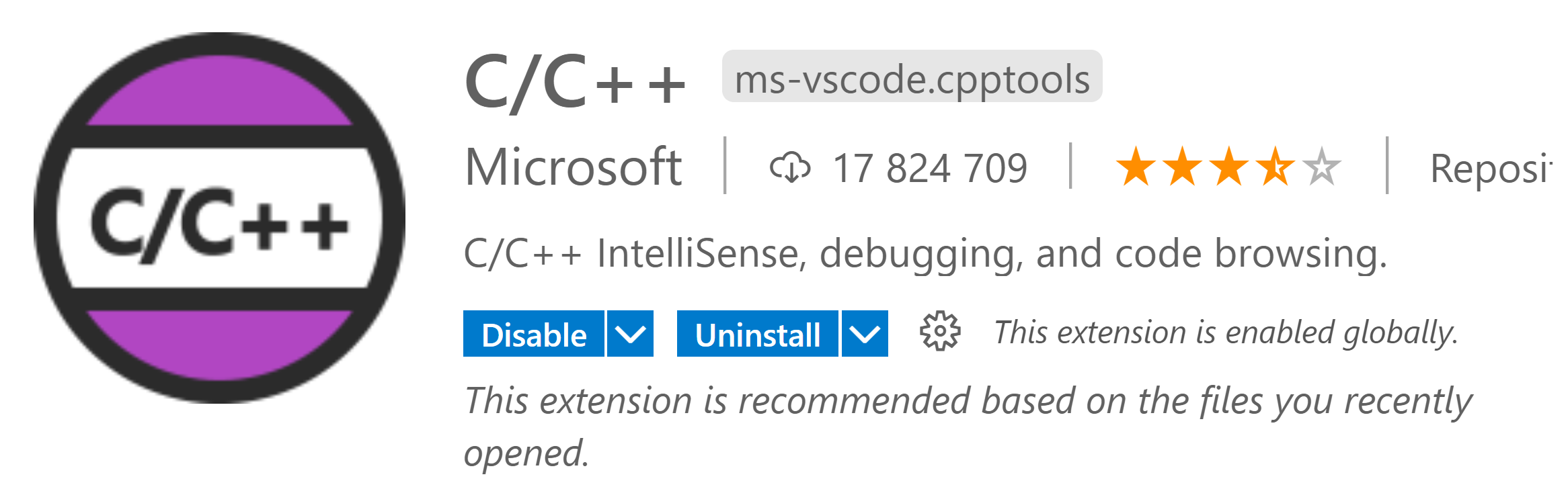
* 1. É criado um ficheiro “*launch.json*”. Alterar

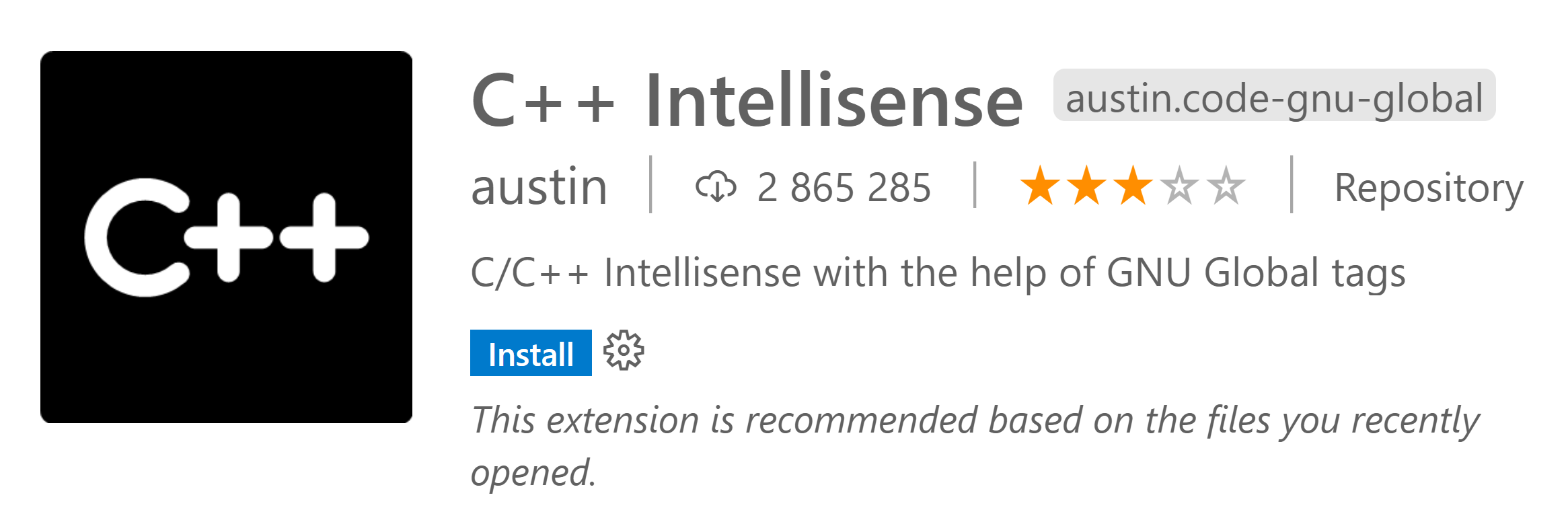
|  |
| --- |
| 1. { 2. // Use IntelliSense to learn about possible attributes. 3. // Hover to view descriptions of existing attributes. 4. // For more information, visit: https://go.microsoft.com/fwlink/?linkid=830387 5. "version": "0.2.0", 6. "configurations": [ 7. { 8. "name": "(gdb) Launch",   Apagar   1. "type": "cppdbg", 2. "request": "launch", 3. "program": "enter program name, for example ${workspaceFolder}/a.exe", 4. "args": [], 5. "stopAtEntry": false, 6. "cwd": "${workspaceFolder}", 7. "environment": [],   Alterar para a pasta do gdb.exe  C:\\Program Files (x86)\\mingw-w64\\mingw32\\bin\\gdb.exe   1. "externalConsole": true, 2. "MIMode": "gdb", 3. "miDebuggerPath": "/path/to/gdb", 4. "setupCommands": [ 5. { 6. "description": "Enable pretty-printing for gdb", 7. "text": "-enable-pretty-printing", 8. "ignoreFailures": true 9. } 10. ] 11. } 12. ] 13. } |

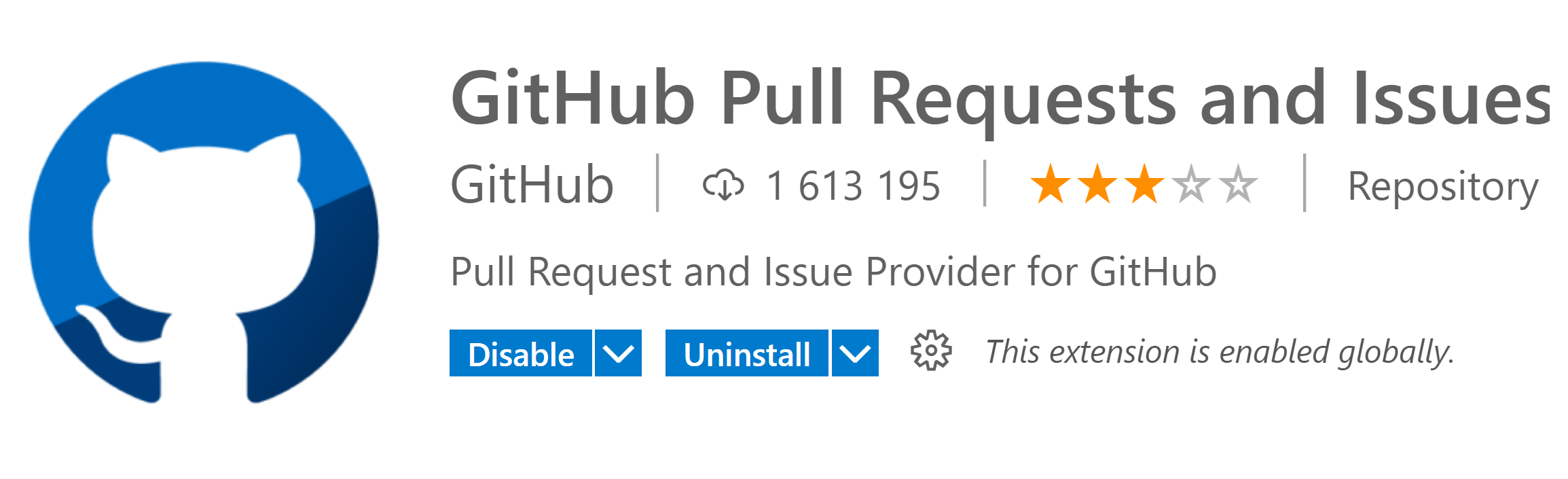
Compilar código para suportar debug

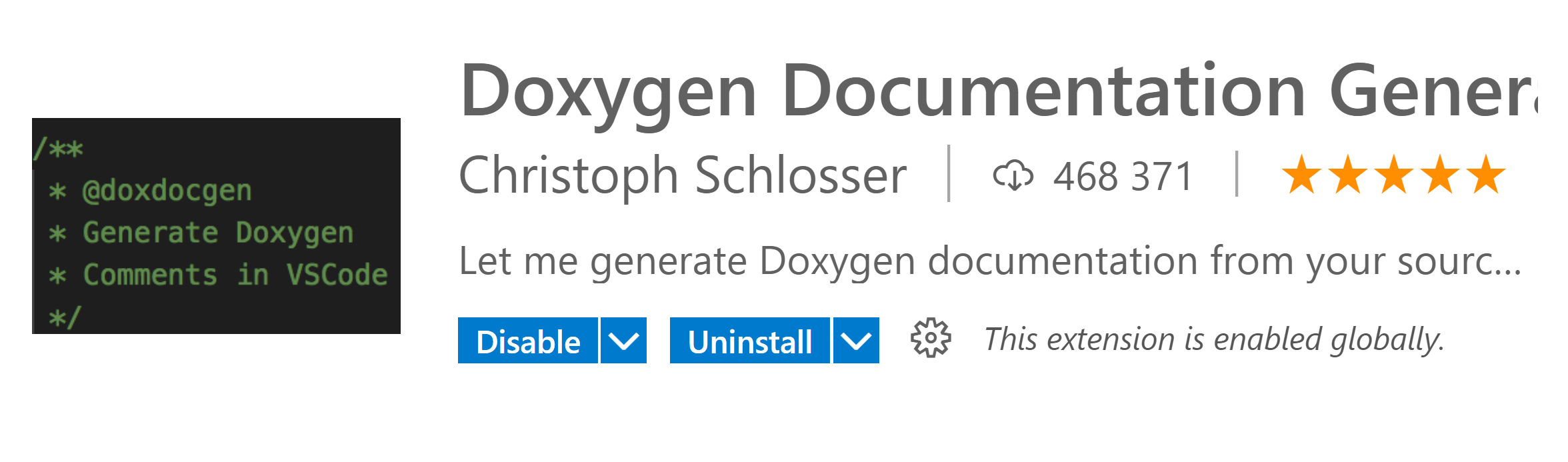
* 1. gcc -Wall \*.c -g

## Extensões a instalar









lufer