Informação orientado para quem quer codar (programar) em C no Vscode

Linguagem C

1. Começando do zero:

\*Esse metodo serve para quem não chegou a instalar o Vscode na maquina, pois se tentou desinstalar e instalar de novo, a maquina guarda informações que ao reinstalar o Vscode, se tiver alguma divergencia ou imprevisto para rodar o codigo, o problema volta. Então sugito que vá para opção 2.

1. Para quem ja instalou o Vscode - quando tiver problema ao rodar o programa em linguagem C

\*Esse metodo é orientado para quem esta tendo problema em rodar o programa C, tera que excluir todas as informações salvas no computadar de configuração feita no Vscode;

\*\* No momento não possuo conhecimento para discorrer e explicar cada problema que aparecer mas o que conheço, vou compartilhar;

\*\*\* Exixste um erro que aparece ao tentar rodar o código por causa do nome do usuario – o erro é: *ld.exe: cannot open output file C:\Users\NomeDoUsuario\Desktop\TestC\hello.exe: Illegal byte sequence*. O *compilador* GCC do MSYS2 tem dificuldade em lidar com acentos e caracteres especiais no caminho. Por se tratar de uma linguagem antiga, as vezes alguns programas antigos não se da bem ao fazer o caminho da pasta em que o nome do usuario contem: letra com acento, “ç”, palavras que contenha espaço, alguns simbolos, etc. Ou seja, as alternativas caso você tenha esse problema é:

* Crie uma pasta em um caminho sem acentos e sem espaço, por exemplo: *C:\Cprojects\TestC,* Coloque seu hello.c lá, abra essa pasta no VSCode e rode novamente;
* Criar uma maquina virtual com nome de usuario comum;
* Criar um novo usuario com outro nome e usar ele para codar o programa – e se quiser pode ate deletar o usuario antigo;
* “Tentar” alterar o nome ao modificar os registros do sistema – se você mudar o nome do usuario na configurações, não acontecera nada, pois o “nome do usuario” pode ate mudar superficialmente mas o nome original esta gravado no sistema mais profundo e alterar esse nome pode causar problemas em alguns programas que ja são dependetes desse “nome original” - vai por mim, eu tentei T.T;
* Formatar a sua maquina – que foi no meu caso T.T

1. Desinstala o Vscode;
2. Apague as pastas de configuração

No Windows, o VSCode guarda configs em duas pastas principais:

* Configurações do usuário (settings, atalhos, extensões instaladas):

C:\Users\SEU\_USUARIO\AppData\Roaming\Code

* Cache e dados temporários:

C:\Users\SEU\_USUARIO\AppData\Local\Code

As extensões ficam em:

C:\Users\SEU\_USUARIO\.vscode

* Basta apagar essas duas pastas.

(Se quiser, pode só renomear para Code\_backup caso queira restaurar depois).

1. Reinstale o VSCode:

Agora, desinstale o VSCode pelo Painel de Controle e depois instale de novo do site oficial: https://code.visualstudio.com/

Com as pastas apagadas, ele vai abrir como novo, sem nenhuma configuração antiga.

1. Instale o MSYS2 (se já tiver, mantenha).

Link: https://www.msys2.org/

Isso instala o compilador gcc e o depurador gdb.

1. Depois rode no terminal do MSYS2 UCRT64 (não o MSYS2 normal):

* pacman -Syu
* pacman -S mingw-w64-ucrt-x86\_64-gcc mingw-w64-ucrt-x86\_64-gdb

1. Configure o Path no Windows para incluir:

* Configurações – sistema – sobre – configurações avançadas do sistema – Variáveis de Ambiente
* Variáveis de usuário para NomeDoUsuario – Path – adicione novo protocolo: C:\msys64\ucrt64\bin
* Variáveis do sistema – Path – adicione novo protocolo: C:\msys64\ucrt64\bin

1. No VSCode, instale apenas a extensão:

C/C++ Extension Pack (da Microsoft, ms-vscode.cpptools).

1. Crie uma pasta C:\Users\SEU\_USUARIO\Desktop\TesteC.

* Dentro dela, crie o arquivo hello.c com este código:

#include <stdio.h>

int main() {

printf("Hello, World!\n");

return 0;

}

1. Abra o terminal do VSCode (Ctrl + `).

Digite:

* gcc hello.c -o hello.exe
* ./hello.exe

1. Se aparecer:

* Hello, World!

Funcionou, compilador + VSCode configurados.

1. Criando arquivo na pasta “.vscode”:

Preparando o arquivo tasks.json e launch.json para você rodar no F5 (sem precisar compilar no terminal. No VSCode, dentro da sua pasta do projeto (TesteC por exemplo), crie uma pasta chamada “.vscode”.

1. Dentro dela, crie dois arquivos:
2. TestC/.vscode/launch.json

{

"version": "0.2.0",

"configurations": [

{

"name": "C Debug",

"type": "cppdbg",

"request": "launch",

"program": "${workspaceFolder}/hello.exe",

"args": [],

"stopAtEntry": false,

"cwd": "${workspaceFolder}",

"environment": [],

"externalConsole": true,

"MIMode": "gdb",

"miDebuggerPath": "C:/msys64/ucrt64/bin/gdb.exe",

"preLaunchTask": "build"

}

]

}

1. Crie um arquivo na pasta - .vscode/tasks.json

{

"label": "build",

"type": "shell",

"command": "gcc",

"args": [

"hello.c",

"-o",

"hello.exe",

"-g"

]

}

1. Aperta “F5” para rodar.
2. versão do tasks.json e launch.json genérica, que funciona para qualquer arquivo .c da pasta, sem precisar trocar o nome do arquivo.
3. Crie um arquivo na pasta - .vscode/tasks.json

{

"version": "2.0.0",

"tasks": [

{

"label": "build active C file",

"type": "shell",

"command": "gcc",

"args": [

"${file}",

"-o",

"${fileDirname}/${fileBasenameNoExtension}.exe",

"-g"

],

"group": {

"kind": "build",

"isDefault": true

},

"problemMatcher": ["$gcc"],

"detail": "Compile o arquivo C atual"

}

]

}

ii) TestC/.vscode/launch.json

{

"version": "0.2.0",

"configurations": [

{

"name": "Debug active C file",

"type": "cppdbg",

"request": "launch",

"program": "${fileDirname}/${fileBasenameNoExtension}.exe",

"args": [],

"stopAtEntry": false,

"cwd": "${fileDirname}",

"environment": [],

"externalConsole": true,

"MIMode": "gdb",

"miDebuggerPath": "C:/msys64/ucrt64/bin/gdb.exe",

"preLaunchTask": "build active C file"

}

]

}