

## set up c++

install vscode

install mingw -> install Cygwin : <https://www.cygwin.com/install.html>

download setup-x86\_64.exe

run in the console on download folder (downloads then cmd, then as admin):

```
setup-x86_64.exe -q -P wget -P gcc-g++ -P make -P diffutils -P libmpfr-devel -P libgmp-devel -P libmpc-devel -P gdb
```

install c ++ on visual studio ide from extension tab

create .vscode folder manually

create tasks.json -> task builds the code using the compiler

```
{
  "version": "2.0.0",
  "tasks": [
    {
      "label": "build", // <- argument is called in the launch.json prelaun. parameter
      "type": "shell",
      "command": "g++",
      "args": ["-g", "*.cpp", "-o", "output"], // name of the input and output file
      "group": {
        "kind": "build",
        "isDefault": true
      }
    }
  ]
}
```

create launch.json -> this debugs the code

```
{
  "version": "0.2.0",
  "configurations": [
    {
      "name": "Debug C++",
      "type": "cppdbg",
      "request": "launch",
      "program": "${workspaceFolder}/output",
      "args": [],
      "stopAtEntry": false,
      "cwd": "${workspaceFolder}",
      "environment": [],
      "externalConsole": true, // if false Unable to perform this action because the
      "MIMode": "gdb",
      process is running in debug
    }
  ]
}
```

```

    "setupCommands": [
      {
        "description": "Enable pretty-printing for gdb",
        "text": "-enable-pretty-printing",
        "ignoreFailures": true
      }
    ],
    "preLaunchTask": "build",
    "miDebuggerPath": "C:/cygwin64/bin/gdb.exe" // <-- Update this path to your actual
gdb.exe
  }
]
}

```

edit:

```
"miDebuggerPath": "C:/cygwin64/bin/gdb.exe" // <-- Update this path to your actual gdb.exe
```

### Debug in vscode with c++

Create main.cpp file

Add:

```

#include <iostream>

int main() {
    std::cout << "Hi!" << std::endl;
    return 0;
}

```

To debug press the play button in the debug tab

To run in the terminal run the executable output file. For example: .\output.exe