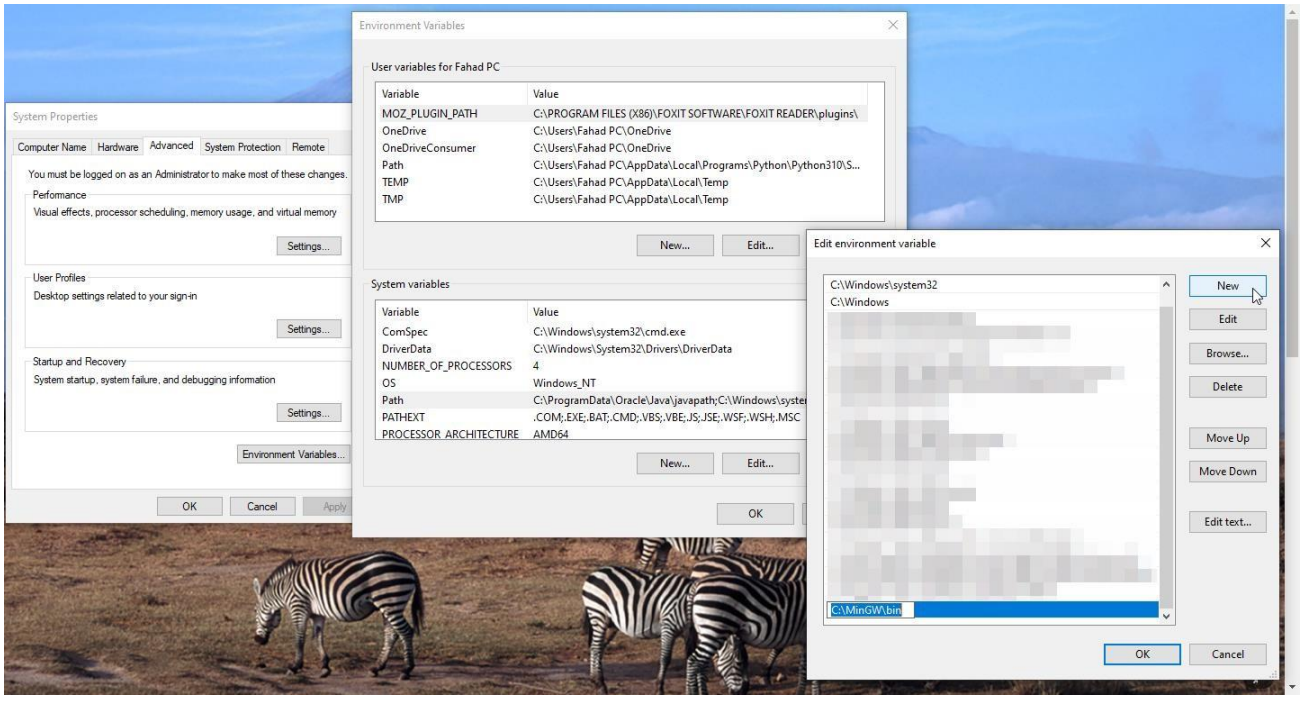
Setup Guide for iGraphics in VS Code

Prepared By: Mustafa Muhaimin Supervised By: Abdur Rashid Tushar

1. Download the old MinGW compiler (MinGW32) from the provided link.
2. Install/Extract and place the MinGW in the C: drive.

   (If you decide to place it elsewhere, then remember and note down the custom location)
3. Place the necessary extra header, library and DLL files as instructed.
   (a) Copy glaux.h and glut.h to {Placed Location}\MinGW\include\GL
   (b) Copy libglut32.a and glaux.lib to {Placed Location}\MinGW\lib
   (c) Check whether glu32.dll and opengl32.dll exists in C:\Windows\System32
4. The MinGW bin folder can be added to the system environment variable path for VS Code to recognize this compiler.

   (This might not be strictly necessary for operation, but it is suggested to do to be safe)
   (a) Search -> Edit the system environment variables -> Environment Variables... ->
   System variables -> Path -> Edit...
   (b) Now either click “New” and input “C:\MinGW\bin” or the path of the MinGW\bin folder if placed in a custom location
   Or you can “Browse...” to the MinGW\bin folder and click “Ok”



5. Now choose a new or existing folder as the project folder.

(a) Put the iGraphics.h, iMain.cpp, stb_image.h and glut32.dll in the project folder

(b) In the project folder, create a new folder named ".vscode"

(c) In the ".vscode" folder, put the tasks.json file

(d) (Optional) If you have placed the MinGW in a custom location, then open the tasks.json file and change the line `"command": "C:\\MinGW\\bin\\gcc.exe",` under the `"label": "Build",` and replace the path with that of the gcc.exe in the custom location of MinGW

be sure to write "\\" in place of any "\" in your path. e.g.

```
{
    "tasks": [
        {
            "label": "Build",
            "type": "cppbuild",
            "command": "F:\\CSE102\\Project\\MinGW\\bin\\gcc.exe",
            "args": [
                "${file}",
```

6. Open the project folder using VS Code:

   (a) Open VS Code

   (b) File -> Open Folder -> Navigate to project folder -> Select Folder

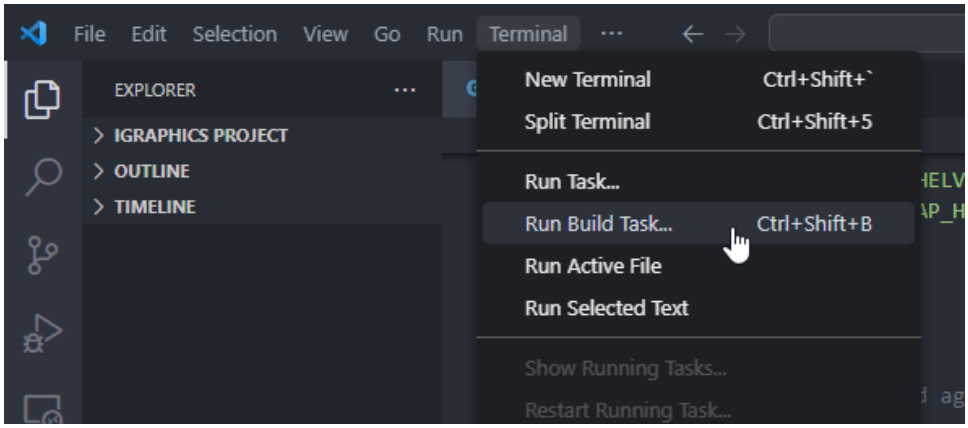
7. VS Code should be set up now for building and running the code.

   Now to build and run your code:

   (a) Open the iMain.cpp file

   (b) Menu bar -> Terminal -> Run Build Task...

   Or press Ctrl+Shift+B



(c) You should see an iMain.exe being built in the project folder without any build errors and afterwards the program run.

Troubleshooting:

If you face any build errors, these are a few steps you can try:

1. Restart VS Code (especially if you were setting up iGraphics with VS Code running)
2. Restart your computer (sometimes a restart is needed for windows to recognize the newly added paths)
3. Double check if the necessary files have been correctly placed in the instructed locations
4. VS Code itself might help you find the problem.
   Click Menu bar -> View -> Terminal or Press Ctrl+` to open the terminal and look in the PROBLEMS tab
   It might show the problem (Especially if it is path related) and how to fix it.
5. One common problem is that - some header file location is specified in iGraphics.h, but it doesn't match the actual path exactly. For example,
   `#include <gl/glut.h>`
   is written, whilst the actual path is C:\\MinGW\include\GL\glut.h
   so it should actually be
   `#include <GL/glut.h>`
6. In the tasks.json file, change the
   `"type": "cppbuild",`
   under `"label": "Build",` to
   `"type": "shell",`
7. Try running VS Code in administrator mode.