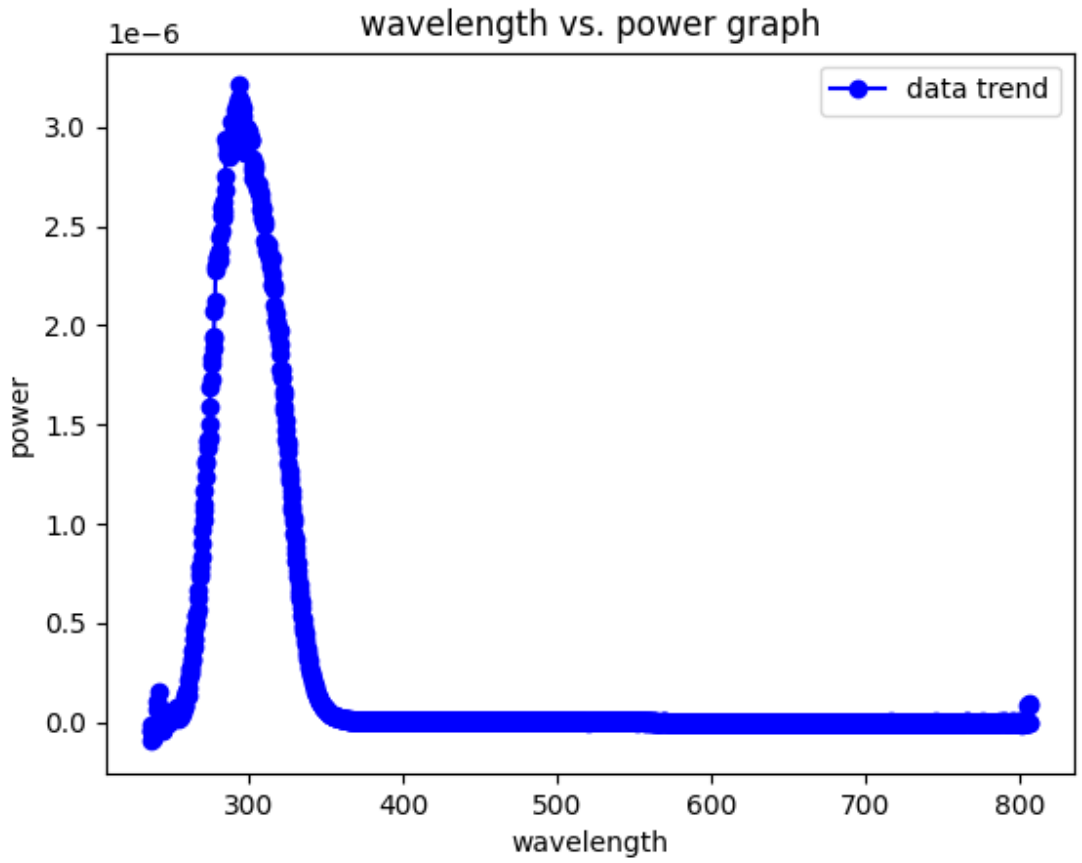
Author: Michelle Liu

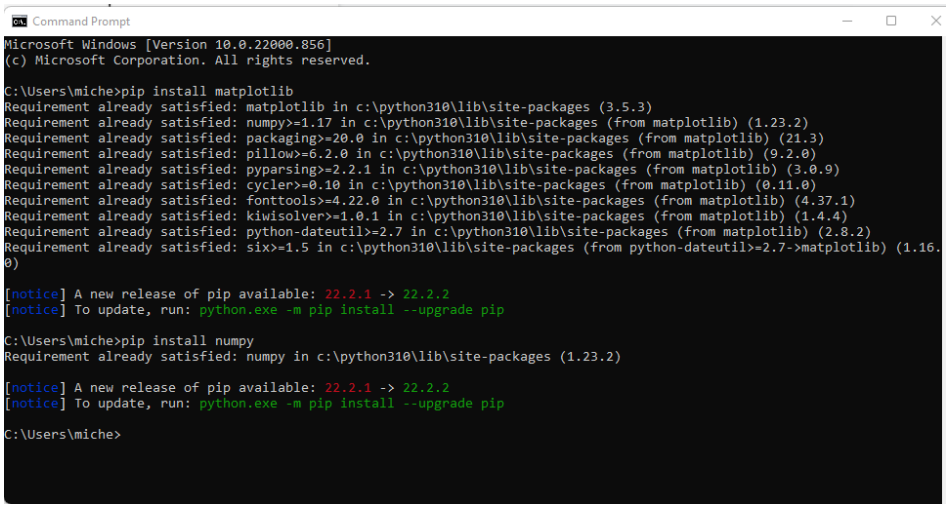
---

I used the wavelength vs. power data (“Vertical0275.ISD”) and generated the following plot. Feel free to use your own data set and follow along.




Now,
My operating system is Windows 11. For softwares, I used the newest Python (Python 3.10), VSStudio Code, and CMake.

**Step 1.** First, I used the Windows built-in Command Prompt and pip to download ‘matplotlib’ and ‘numpy’ libraries.

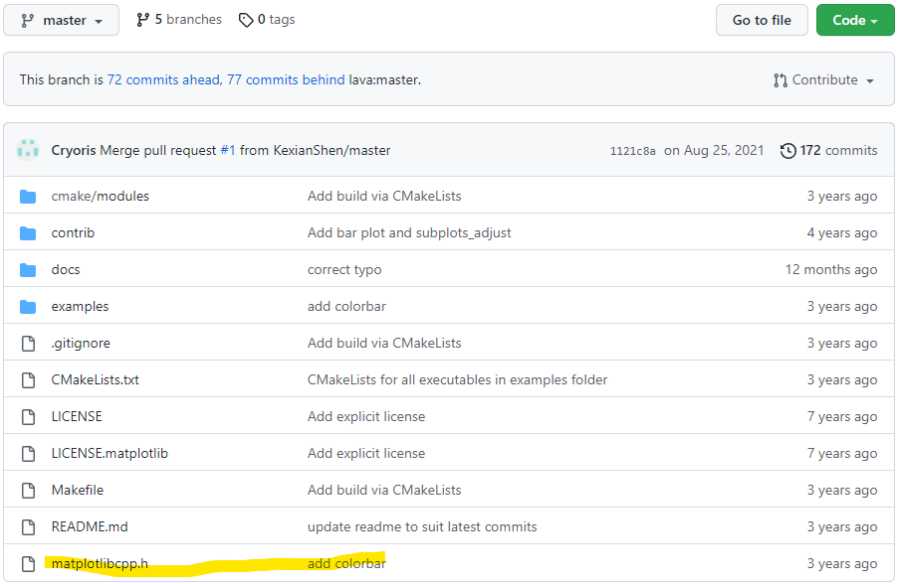



Note here my libraries are installed correctly and running this code again will show you the corresponding paths. Save them up as these will be useful later.

**Step 2.**
Next up, time to create the scripts. Make sure there is a dedicated folder for the project and include the "Vertical0275.ISD" file from the Lab07 file on the course website. Also, make sure VS Studio is in the correct directory.

I used Virtual Studio Code and created three scripts: "makefile.cpp", "matplotlibcpp.h", "main.cpp".

1). In "matplotlibcpp.h", copy and paste the matplotcpp.h code from the matplotcpp github.




2). Now, in "main.cpp", we will include the headers and namespace.

```cpp
main.cpp > ...
1   #include "matplotcpp.h"
2   #include <iostream>
3   #include <fstream>
4
5   namespace plt = matplotlibcpp;
6   using namespace std;
7
```

3). Next up, we need to figure out how to actually compile our main file to avoid any error. This part was particularly tricky since c++ will fail if the paths are incorrect.

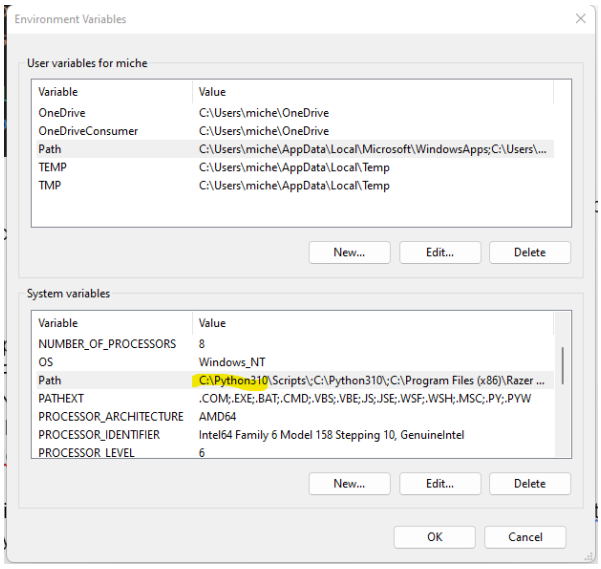
Here is the command I ran:
g++ main.cpp -I c:\Python310\include -I c:\python310\lib\site-packages\numpy\core\include -L C:\Python310\libs -lpython310

**There are 5 parts:**

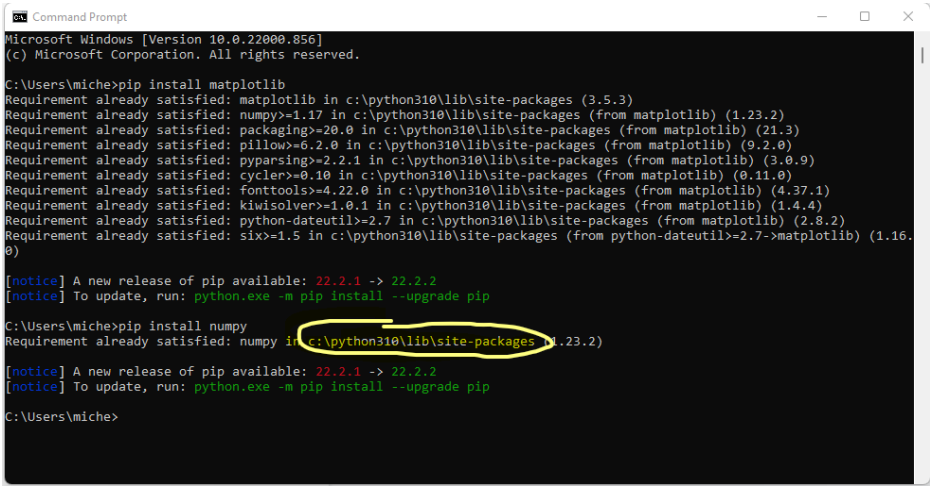
**1).** g++ main.cpp
**2).** -I PATH\TO\PYTHON**\include**
**3).** -I PATH\TO\NUMPY**\numpy\core\include**
**4)**. -L PATH\TO\PYTHON**\libs**
**5)**. -lpythonversion

For this part, create a simple int main to generate a simple plot. We will use this to test the command runs correctly.

For 2). I had to manually reinstall Python and select the “Add to PATH” option. Next in the Windows built-in “Edit- Environment Variables” locate the Path to Python and save the path into “makefile.cpp”. Remember the “\include” at the end.



For 3). This is the path to the “numpy” library installed using “Command Prompt”.




For 4). This is the path to the python libraries. I included all libraries for simplicity here.

This part’s command is the path to Python and **\libs**.

Once the commands are correctly input, the program should be able to create plots now!

**Step 3:**

Write “main.cpp” so that it generates the plot from the data.

I wrote two functions: get_wavelength_data() and get_power_date().
The two functions take in an ifstream by reference and return a vector of doubles corresponding to wavelength/power.

Once these two functions are written correctly, we should be able to create an int main() to generate our plot.

Use ifstream to read in the data. Next up, use get_wavelength_data(inF) and store this inside a vector of doubles (I called it ‘x’). Use get_power_data(inF) and store this inside a different vector of doubles (‘y’).

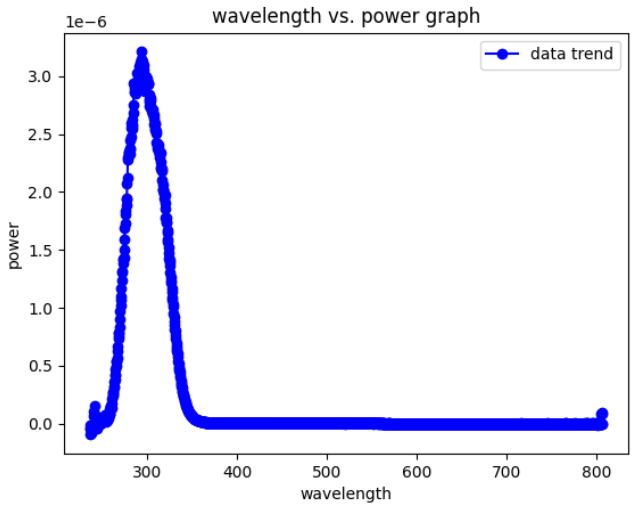
Then, use matplot functions to create the script for plotting.

```
plt::figure();
plt::plot(x,y, "bo-", {{"label", "data trend"}});
plt::xlabel("wavelength");
plt::ylabel("power");
plt::title("wavelength vs. power graph");

plt::legend();
plt::show();

return 0;
```

Now you should be able to compile “main.cpp” and generate the following plot.




Note it will show up as “a” inside the project file and opening this file will show the plot.