C02

Part 1:

I already had Jupyter on my laptop, so the only thing I had to do to get this working was to compile and run the C program:

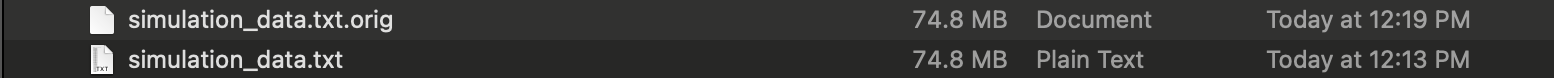
A screen shot of a computer

Description automatically generated

Then open and run the Python program in Jupyter Labs:A screenshot of a screen

Description automatically generated

Both versions of the txt file:



Part 2:

The shallow-water-advanced.c file was edited to look like this:

A computer screen shot of a program

Description automatically generated

The file was changed to have only the main method, as well as any definitions and include statements for the program to work.

The utilities.c file looks like this: A screenshot of a computer program

Description automatically generated

It again has any necessary definitions and include statements fort the program to work, as well as any function definitions.

The utilities.h file looks like this:

A screenshot of a computer

Description automatically generated

This file has any necessary include statements and definitions, as well as the function prototypes for the program.

All the files compiled correctly with the commands given in the assignment, and the program was able to be run successfully. After using the diff command on the two files, the original program and the new one produce the same output:

A screen shot of a computer

Description automatically generated

Part 3:

In order to get the program to take command line arguments, I instead created global variables for NX, NY, timesteps and FILENAME to be able to access them in the header file as well as the main c file. The variables are defined in the main method, then can be used in the header file as the variables passed into the functions defined there. The three edited files are below:

shallow-water-advanced.c

A screenshot of a computer program

Description automatically generated

utilities.c

A screenshot of a computer program

Description automatically generated

utilities.h

A screenshot of a computer

Description automatically generated

After compiling the files and running the program to get the new output file, and running the diff command on them, we see the new program outputs the same file when using the inputs as given in the original.

A screen shot of a computer

Description automatically generated

This is the output when using 70x70 inputs:

A yellow circle with a blue background

Description automatically generated

The output when using 30x30 as the inputs:

A screenshot of a computer screen

Description automatically generated

The output when using 20x20 as inputs:

A screenshot of a computer screen

Description automatically generated

After a certain amount of time, the simulation turns into a white square, which I assume happens because there is not enough memory being allowed for the simulation.