# LAB 8: DISTRIBUTED COMPUTING

FALL 2024

## REQUIRED SKILLS & KNOWLEDGE

1. Strings
2. Functions
3. JSON & cURL
4. Distributed Data

## ASSIGNMENT OBJECTIVE

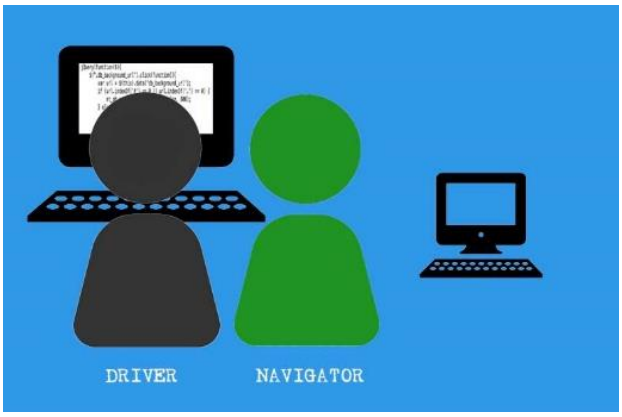
You will be able to develop and implement algorithmic solutions for interacting with a remote service.

## DESCRIPTION

You are given a program that is incomplete. Your job in this assignment is to complete a C++ program that will read data from a remote data repository, modify the given data, and then display the data to the screen.

## PAIRED PROGRAMMING OPTION

You may complete this lab assignment alone or you have an OPTION to complete this lab with a lab partner using paired programming techniques. If you choose to pair program, follow the directions in this section. Your first step is to **exchange preferred contact information** just in case you are unable to complete the lab during lab and need to meet outside of lab class to finish.




### SUBMISSION IN ILEARN

You will both upload the same exact zip file to your Lab 3 assignment in ilearn. Each source file should have both of your names in the comment block at the top. Both students will receive the same feedback and grade.

### HOW TO PAIR PROGRAM

One of you can start writing (or debugging) the initial code (DRIVER) while the other reviews and suggests improvements (NAVIGATOR). **Take turns regularly (every 10 to 15 minutes)** to ensure both of you are actively involved.

## WARNING ABOUT ACADEMIC MISCONDUCT

Programming assignments in this course are your exams and should be only YOUR work.

- **You are not allowed to get help from friends, family, or other students.**
- **You are also not allowed to use generative AI such as ChatGPT or CoPilot to assist in writing your code.**

If I suspect generative AI was used in your code, I will fill out a charging document to charge you with academic misconduct. If the charge stands after review from the Academic Misconduct Committee, you will receive a zero for your grade and this charge will be on your permanent student record. Being charged with academic misconduct more than once can result in expulsion from Tennessee Tech. This is a very serious matter, which is why I am stressing it in several places. In more advanced CSC courses, using generative AI as a tool is encouraged, but in this course where you are learning foundational aspects of coding and problem solving, it can't be used.

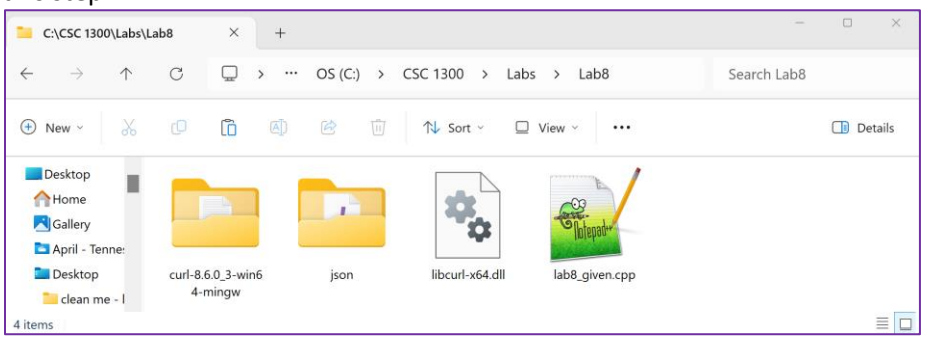
# INSTRUCTIONS

## REQUIRED SETUP

1. Use your **Lab 8** folder for this assignment.

2. Download the following files from ilearn and place it in this folder:
   a. curl-8.6.0_3-win64-mingw
   b. json
   c. lab8_given.cpp

2. Copy the file named **libcurl-x64.dll** that is inside the **curl-8.6.0_3-win64-mingw** folder and paste it directly inside the **Lab 8** folder. See screen capture below that shows what the **Lab 8** folder will look like after you have done this step.




## NEW C++ LIBRARIES YOU WILL USE

- **`#include <curl/curl.h>`**
  This library is required to use curl. Curl is used to transfer data specified with URL syntax, which you will need in this program since you are accessing and transferring data from an HTML page.

- **`#include <nlohmann/json.hpp>`**
  The data you will be accessing is in JSON format, which is why you need this library to be able to access and read the data with your program. The JSON for Modern C++ library was created by Mr. Niels and is available at https://github.com/nlohmann. The README file for his JOSN library is here: https://github.com/nlohmann/json/blob/develop/README.md

- **`#include <sstream>`**
  You may need the sstream library to be able to define a variable of type ostringstream. This is part of the given code and is required to use curl to transfer the data to your program.

## JSON INFORMATION

JSON text is very readable by both humans and computers. Below is an image of a JSON object. JSON objects contain key-value pairs. For example, the key “place” has value “Volcano Islands, Japan region”.


```
"properties": {
  "mag": 5.1,
  "place": "Volcano Islands, Japan region",
  "time": 1729792288383,
  "updated": 1729795339453,
  "tz": null,
  "url": "https://earthquake.usgs.gov/earthquakes/eventpage/us7000nmyu",
  "detail": "https://earthquake.usgs.gov/earthquakes/feed/v1.0/detail/us7000nmyu.geojson",
  "felt": 1,
  "cdi": 2.7,
```

## PROGRAM SPECIFICATIONS

1) Rename the given source file, **lab8_given.cpp**, to **lab8.cpp**.

2) Look at the **JSON** file that you will be accessing with your code. The URL is https://earthquake.usgs.gov/earthquakes/feed/v1.0/summary/all_month.geojson (taken from https://earthquake.usgs.gov/earthquakes/feed/v1.0/geojson.php)

3) Add the new **`#includes`** mentioned above in this assignment (curl, json, sstream).

4) In the main function of your source code there are four function calls to **`curl_easy_setopt()`**.
   a) The first call to this function contains the URL of the website that you are accessing.
   b) Make sure this argument is `"https://earthquake.usgs.gov/earthquakes/feed/v1.0/summary/all_month.geojson"` (including the double quotes).

5) You should not have to modify any of the other code in the **`if(curl)`** block of code.
   a) This code will use curl to transport the data from the website to your local machine and will store the data in an **`ostringstream`** variable called **`oss`**, which is then changed to a string and stored in a string variable called **`readBuffer`**.

6) The **main** function in this program contains C++ exception handling with keywords **`try`** and **`catch`**.
   a) The **`try`** statement allows you to define a block of code to be tested for errors while it is being executed.
   b) The **`catch`** statement allows you to define a block of code to be executed if an error occurs in the **`try`** block.
   c) The **`catch`** statement in this program prints the exception (error) to the screen. If you can successfully read from the HTML file, your program should only execute the **`try`** block of code.
   d) For more information on **`try`** & **`catch`**, visit https://www.w3schools.com/cpp/cpp_exceptions.asp

7) A range-based for loop (foreach loop) is already given to you. Make sure that your loop matches the one given below. This is a type of loop only used with variables that contain multiple data pieces, such as our jsonObject (and in Module 8 we will use it with arrays & vectors). What this loop will do is iterate through (go through) each piece of data in the jsonObject and store the object in "element".

```
for (auto& element : jsonObject["features"])
```

   a) As mentioned before, JSON objects contain key-value pairs. Your job is to:
      i) **print the first 10 earthquakes that have a magnitude of 4.5 or higher**
      ii) **print the earthquakes that caused a tsunami**

   b) To help you with the syntax of the code, you are given the first foreach loop. You will just need to add the other loop to complete the lab. The **`if`** statement in the first loop should look like:

```
if (element["properties"]["mag"] >= 4.5)
```

   c) Be sure to look at https://earthquake.usgs.gov/earthquakes/feed/v1.0/geojson.php to see how the JSON is formatted.

## COMPILATION

### MAC USERS:

You may either compile with this statement (make sure if you copy/paste to retype the quotations):

```
g++ lab8.cpp -Wall -std=c++17 -I "/your/path/here/curl-8.6.0_3-win64-mingw/include" -I "/your/path/here/json/include" -o lab8_run -lcurl
```

Or you may download the *Makefile* that is given – just be sure to update your include path.

### WINDOWS USERS:

**You may either compile with this statement** (make sure if you copy/paste to retype the quotations)**:**

```
g++ lab8.cpp -Wall -std=c++17 -I "/your/path/here/curl-8.6.0_3-win64-mingw/include" -I "/your/path/here/json/include" -L "/your/path/here/curl-8.6.0_3-win64-mingw/lib" -o lab8_run -lcurl
```

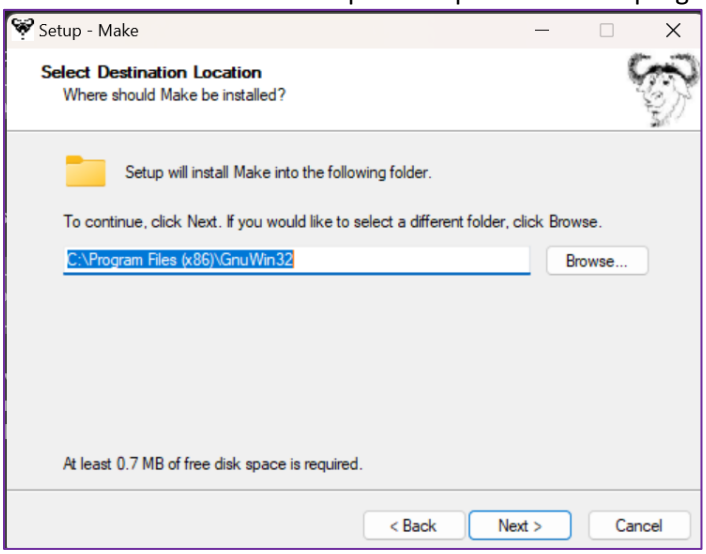
**Or you may download the *Makefile* that is given by following the directions below:**

1. You will need to update your include paths in your *Makefile*.

2. *Make* may not be recognized as a command so you will need to do the following:

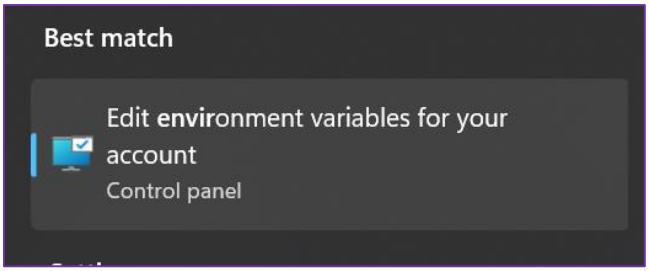
   a. Go to https://gnuwin32.sourceforge.net/packages/make.htm and download the Setup Program. When you click on the download, it will take you to the setup wizard

   b. Make note of where the setup wizard puts the Make program

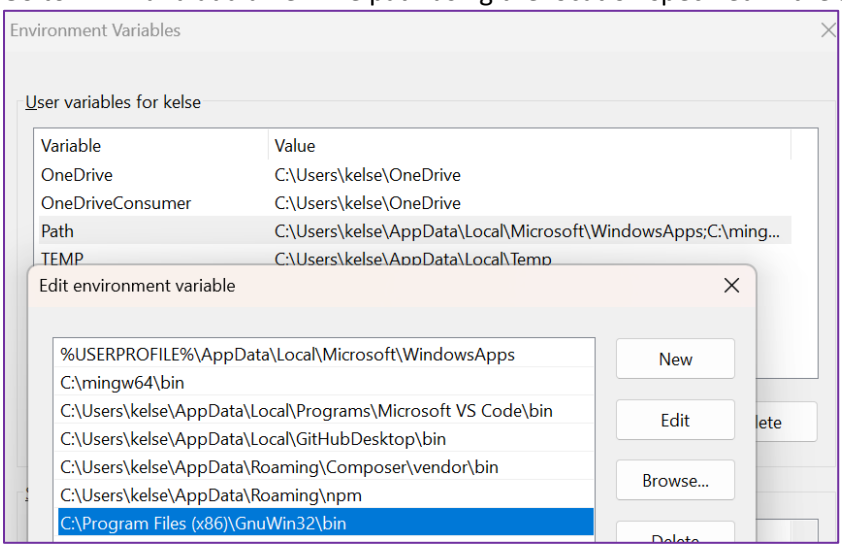



   c. Get through the rest of the setup wizard.

   d. Go to edit environmental variables



e. Go to PATH and add a new file path using the location specified in the setup wizard /bin




## MAKEFILE:

The commands to run the Makefile are given in the file as well as below:

***make:*** builds the executable file (equivalent to running your g++ statement)

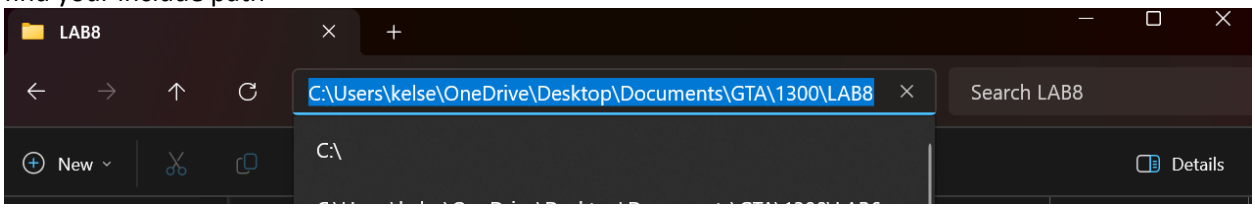
***make run:*** runs your executable file

***make clean:*** removes the executable file from your directory

# FILE PATHS

## WINDOWS:

To find your include path, navigate to your lab 8 directory in your File Explorer. Then, highlight the bar at the top to find your include path




## MAC:

To find your include path, navigate to your Lab 8 directory in the terminal. Then, type in the pwd command, which will return your path (remember, ls to print what is in your current directory and cd to navigate through the directories)

```
acrockett2@BRUN238-CO-M04 ~ % ls
Desktop         Downloads       Movies          Networkshares   Public
Documents       Library         Music           Pictures
acrockett2@BRUN238-CO-M04 ~ % cd "Documents/CSC 1300/Labs/lab8"
acrockett2@BRUN238-CO-M04 lab8 % pwd
/Users/acrockett2/Documents/CSC 1300/Labs/lab8
acrockett2@BRUN238-CO-M04 lab8 %
```

# SAMPLE OUTPUT

There is no user input in this program. If you successfully retrieved the data and printed it to the screen, it should look like this when you run it (it may be different since the data is updated daily):

```
Earthquakes with magnitude greater than or equal to 4.5:

Earthquake 1:
        Place: "139 km NW of Neiafu, Tonga"
        Magnitude: 4.9
Earthquake 2:
        Place: "145 km SSW of Severo-KurilГ‡Ösk, Russia"
        Magnitude: 6.2
Earthquake 3:
        Place: "Volcano Islands, Japan region"
        Magnitude: 4.7
Earthquake 4:
        Place: "West Chile Rise"
        Magnitude: 5
Earthquake 5:
        Place: "Izu Islands, Japan region"
        Magnitude: 4.9
Earthquake 6:
        Place: "5 km W of Cruz Grande, Mexico"
        Magnitude: 4.9
Earthquake 7:
        Place: "off the coast of Central America"
        Magnitude: 4.6
Earthquake 8:
        Place: "south of the Fiji Islands"
        Magnitude: 5.5
Earthquake 9:
        Place: "49 km SSW of Labuha, Indonesia"
        Magnitude: 4.9
Earthquake 10:
        Place: "56 km SSE of Kawalu, Indonesia"
        Magnitude: 4.9

Earthquakes that caused a tsunami:

Earthquake 1:
        Place: "235 km SE of Akutan, Alaska"
        Magnitude: 5.3
Earthquake 2:
        Place: "32 km W of Akhiok, Alaska"
        Magnitude: 4.4
Earthquake 3:
        Place: "175 km SW of Nikolski, Alaska"
        Magnitude: 5
Earthquake 4:
        Place: "5 km SE of Ontario, CA"
        Magnitude: 3.99
Earthquake 5:
        Place: "81 km SSW of Unalaska, Alaska"
        Magnitude: 5.4
Earthquake 6:
        Place: "125 km SE of Neiafu, Tonga"
        Magnitude: 6.6
Earthquake 7:
        Place: "3 km  of Aromas, CA"
        Magnitude: 4.15
```

## FILL OUT THE LAB REPORT

You will fill out this lab report for every lab and it is part of your grade. To get credit, you must upload a screenshot of the confirmation page to this lab assignment. Name your screenshot **lab8ReportProof**.
**Lab Report Link:** https://tntech.co1.qualtrics.com/jfe/form/SV_d6BGc6kzQdSvBmS

## WHAT TO TURN IN

Create a zip file named **labPartner1username_labPartner2username_lab8** containing the following .cpp files and upload it to ilearn. Replace labPartner1username with one lab partner's TTU username and replace labPartner2username with the other lab partner's TTU username. Example: **`jdean42_acrockett43_lab8.zip`**

- Lab8.cpp
- Lab8ReportProof

Remember, both lab partners should upload this zip file to their ilearn assignment.