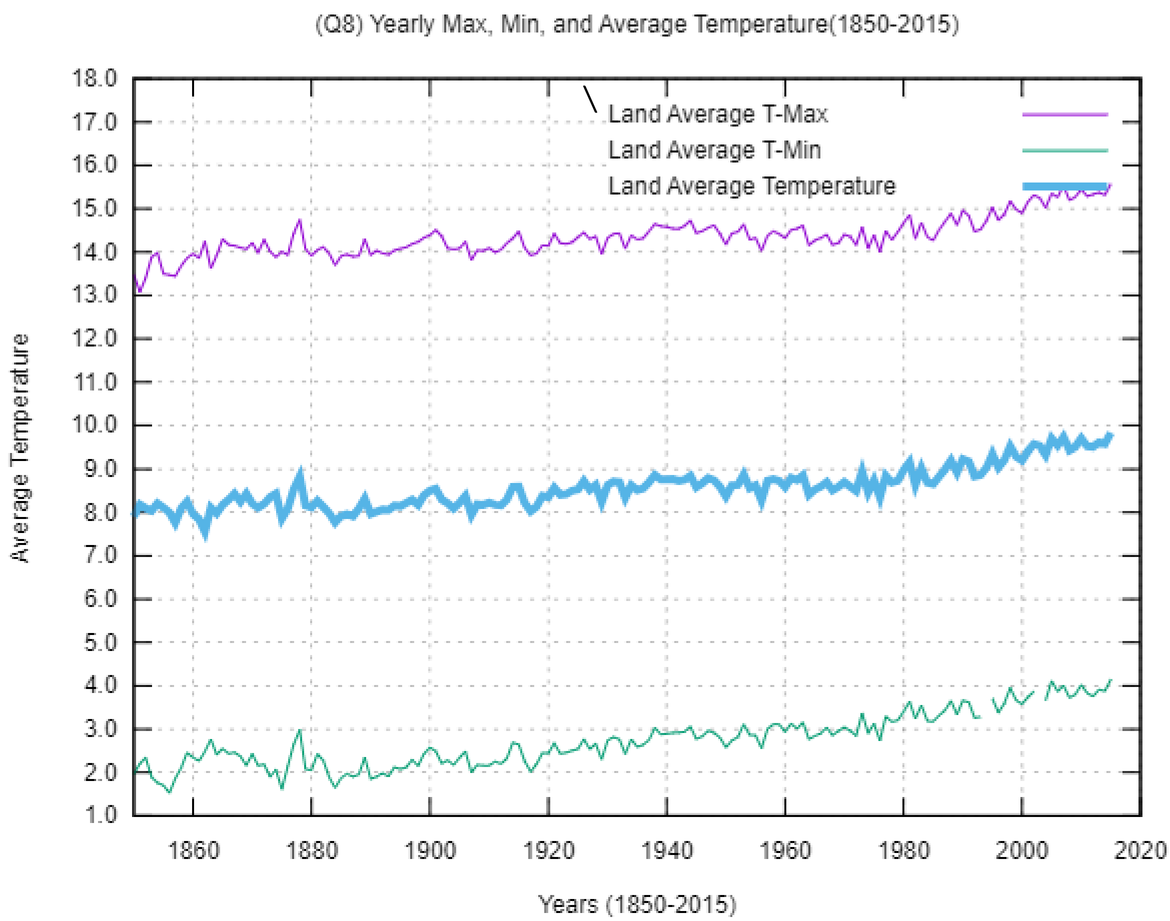
Visualizing Climate Change

(through C-Programming and GNU plot)



Introduction:

The purpose of this project is to analyze, calculate, and make conclusions regarding the real world weather data which was collected from various monitoring systems worldwide. The following dataset contains the monthly land and oceans temperatures starting from 1750 to 2015. The aim for this project is to import this data set into a C program (which the code is written on the text editor Geany) and perform calculations, generate graphs, and draw conclusions from the output and graphical data. The following graphs will be generated through the use of GNUPlot which is a portable command line graphical web application that requires the output of a .txt file (generated from the C program) and its built-in functions also known as the GNUPlot scripts. The project will be presented as a single, easy-to-read program that is divided into several sections which address the specific analytical questions. Additionally the project is intended to help climate change researchers and students who are interested in analyzing historical weather data.

The analytical questions include calculating yearly averages from 1760 to 2015, determining the average land temperature for different centuries, finding monthly averages from 1900 to 2015, identifying the hottest and coldest recorded months and years, and generating various types of graphs in order to visualize the data effectively. The C program will contain syntax and concepts which have been taught in class such as the use of string manipulation, structures, pointers, conditional statements such as the switch cases and nested loops, importing, reading and parsing files. In addition, the GNUPlot scripts will be implemented to show the visual representation of graphs which include line plots, bar plots, error bar plots ensuring that all labels, legends and titles are automatically generated through the scripts. The project would succeed in providing a concise insight regarding the historical weather trends and further analyze a deeper understanding of climate/temperature patterns over centuries.

Algorithm Explanation:

Part 1:

Based on the land average temperature column, calculate the yearly averages for each year between 1760 and 2015 (the average of the twelve months of each year). One average per year. Ignore the years 1750-1759.

To calculate yearly average temperatures from 1760 to 2015, we begin by reading temperature data from the input file. For each line of data, we extract the year, month, and land average temperature. Then, we accumulate the monthly temperatures for each year and divide the sum by 12 to obtain the yearly average temperature. These yearly averages are written into an output file along with the corresponding year. We repeat this process for all years within the specified range, ignoring data before 1760. Finally, we close both the input and output files.

Output-table:

First 10 years

1760	7.185167
1761	8.772500
1762	8.606500
1763	7.496750
1764	8.400333
1765	8.251917
1766	8.405667
1767	8.221500
1768	6.781333
1769	7.694583
1770	7.691917

Last 10 years

2005	10.246250
2006	9.532500
2007	9.732167
2008	9.431750
2009	9.505250
2010	9.703083
2011	10.987000
2012	10.313667
2013	9.606500
2014	9.570667
2015	9.831000

Part 2:

Based on the land average temperature column, calculate the average land temperature for the different centuries: 18th century (1760-1799), 19th century (1800-1899), 20th century (1900-1999) and 21st century (2000-2015). One average per century.

To find the average land temperature for different centuries (18th, 19th, 20th, and 21st), we iterate through the temperature data from the input file. For each year, we calculate the yearly average temperature as done in Question 1. Then, based on the year range, we gather the yearly averages for each century. After collecting the yearly averages for the entire century, we divide the sum by the number of years in that century to obtain the century average temperature. This process is repeated until all years are processed. Finally, we print out the average temperatures for each century as output.

Output-table:

The average land temperature(s) for the 4 centuries:

```
The 18th century average was: 8.215.
The 19th century average was: 8.009.
The 20th century average was: 8.638.
The 21st century average was: 9.542.
```

Part 3:

Based on the land average temperature column, calculate the monthly averages for each month for all years combined between 1900 and 2015. A total of twelve averages. One average per month.

To compute monthly average temperatures for all years between 1900 and 2015, we parse the temperature data from the input file. For each line, we extract the year, month, and land average temperature. We maintain an array to store the sum of temperatures for each month. As we iterate through the data, we add the land average temperature to the corresponding month's sum. At the end of the data range, we divide each monthly sum by the total number of years to obtain the monthly average temperature. These monthly averages are then printed out for each month on the output screen.

Output-table:

The monthly averages for each month for all years combined between 1900 and 2015

```
For month 1 the average was 2.815.

For month 2 the average was 3.331.

For month 3 the average was 5.396.

For month 4 the average was 8.537.

For month 5 the average was 11.395.

For month 6 the average was 13.540.

For month 7 the average was 14.449.

For month 8 the average was 13.958.

For month 9 the average was 12.167.

For month 10 the average was 9.527.

For month 11 the average was 6.223.

For month 12 the average was 3.812.
```

Part 4:

Based on the land average temperature column, what was the hottest month recorded and what was the coldest month recorded (month and year in each case). In case of a tie, mention only one (doesn't matter which one).

In determining the hottest and coldest months recorded, we iterate through the temperature data from the input file. For each line, we extract the year, month, and land average temperature. We compare the current temperature with the recorded hottest and coldest temperatures. If the current temperature exceeds the previous hottest or coldest, we update the respective month and temperature details. This process continues until all data is processed. Finally, we have the details of the hottest and coldest months as the output.

Output-table:

Land Hottest month and coldest month based on average (month & year)

```
The hottest month was 1761/07 at 19.021.
The coldest month was 1768/01 at -2.080.
```

Part 5:

Based on your answer in question 1, what year was the hottest and what year was the coldest?

In identifying the hottest and coldest years based on yearly average temperatures, we follow a similar process as in Question 1 to compute yearly averages. As we calculate yearly averages, we compare each year's temperature with the recorded hottest and coldest temperatures. If a year's average temperature exceeds the previous hottest or is lower than the previous coldest, we update the respective year and temperature details. This process repeats until all years within the specified range are processed. Finally, we print out the details of the hottest and coldest years.

Output-table:

Land hottest year and coldest year from 1760 and 2015 based on question 1

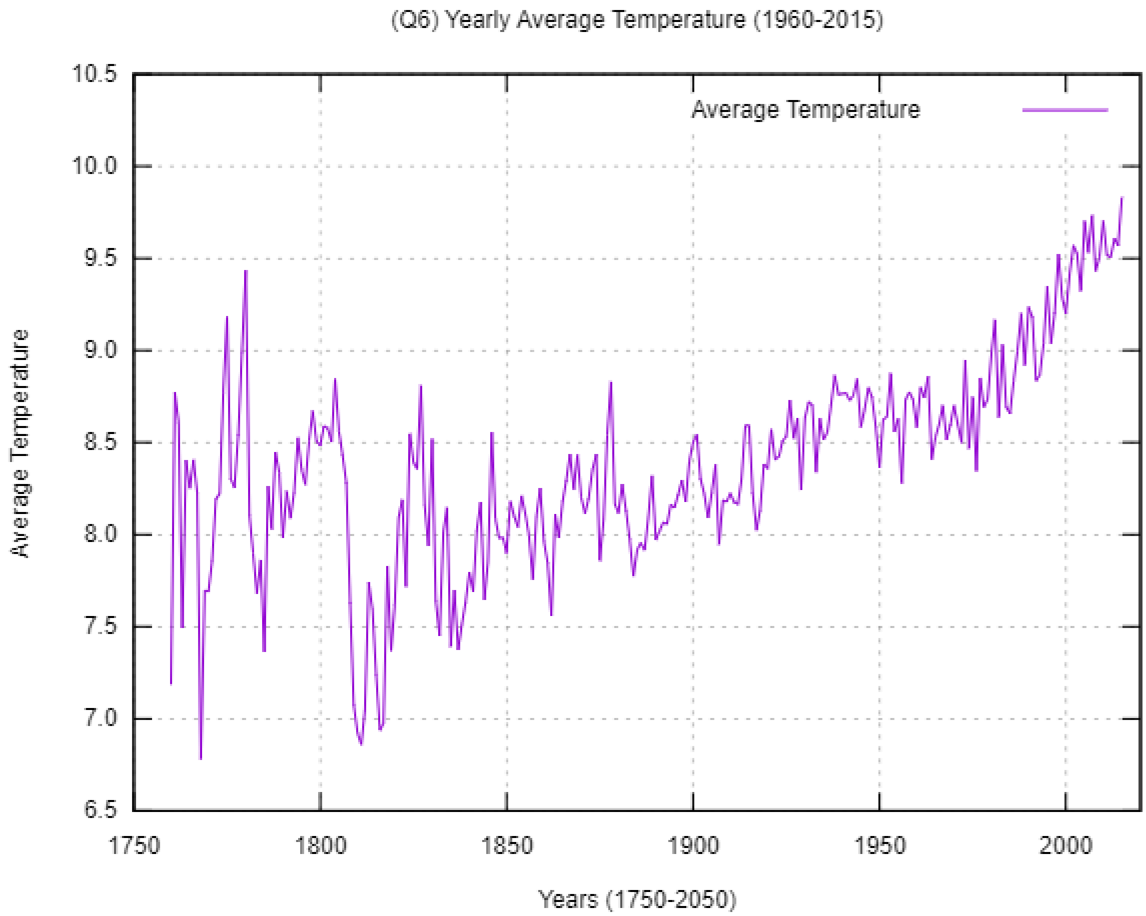
```
The hottest year was 2015 at 9.831.
The coldest year was 1768 at 6.781.
```

Part 6:

Based on your answer in question 1, generate a GNUPlot data file and use GNUPlot to make a graph (line plot) of the yearly temperatures for the years 1760 to 2015. Label the axes clearly and add a title and legend to your graph.

To generate a graph depicting yearly temperatures from 1760 to 2015 using GNUPlot, we create a data file containing the yearly temperatures calculated in Question 1. With GNUPlot, we plot the data file using a line plot. We ensure to label the axes clearly, add a descriptive title, and include a legend indicating the data represented. Once the graph is generated, we display it for visualization.

GNUPlot diagram:



The graph displays a mixed pattern of upward and downward fluctuations in yearly average land temperatures from 1760 to 2015, amidst a dominant upward trend after 2000. However, within this trend, there are fluctuations or variations in temperature from year to year. These fluctuations could be influenced by various factors such as natural climate variability, oceanic oscillations, and human-induced emissions. Despite these short-term fluctuations, the overall trend clearly shows a warming pattern, highlighting the long-term impact of climate change.

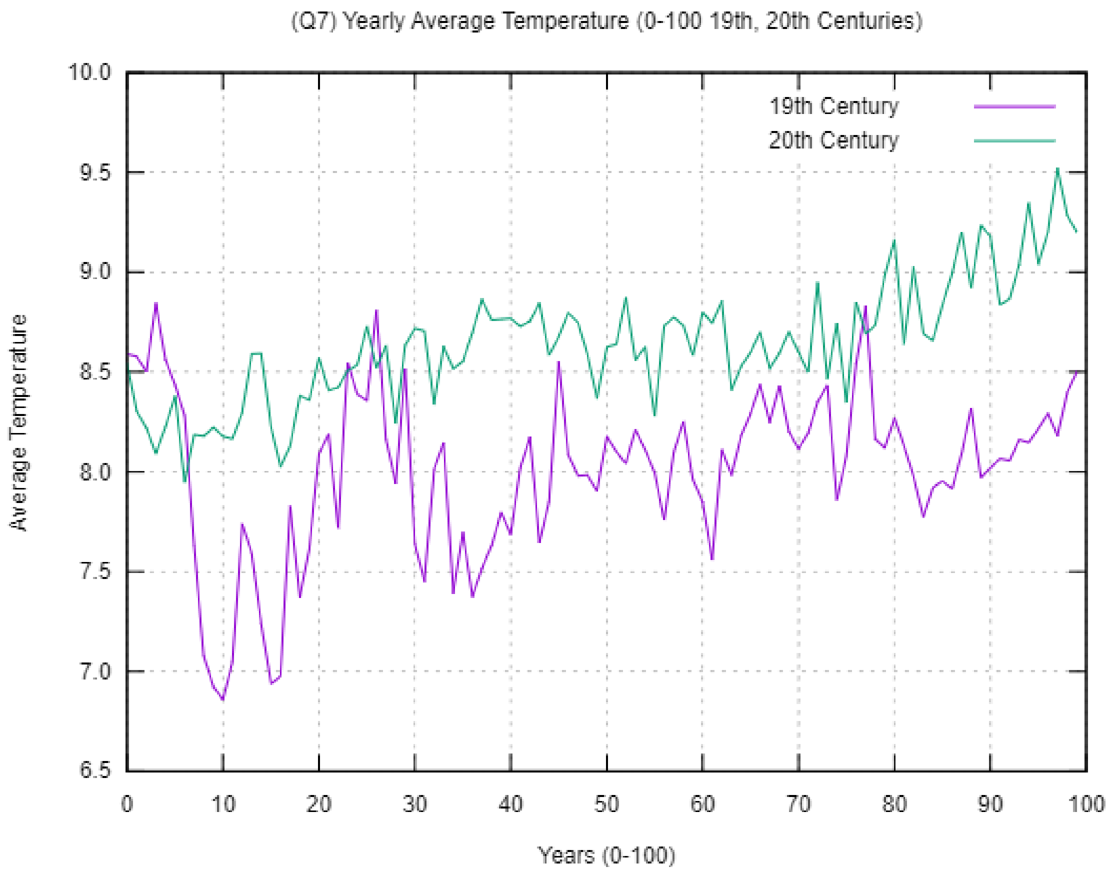
Part 7:

Generate a GNUPlot data file and use GNUPlot to make a graph (line plots) of the average land temperatures for the 19th and 20th centuries. Put both lines on the same figure. Ensure that you have the same x-axis scale (for example 1852 and 1952 would both have an x-value of 52). Have your two line plots with different colors. Label the axes clearly and add a title and legend to your graph.

For plotting the average land temperatures of the 19th and 20th centuries on the same figure using GNUPlot, we create separate data files for each century's temperature averages. Then, in GNUPlot, we generate line plots for both centuries, ensuring the x-axis scales are consistent. We assign different colors to distinguish between the lines representing the 19th and

20th centuries. Additionally, we label the axes clearly, add a descriptive title, and include a legend for clarity. After generating the plot, we proceed to present it on the screen.

GNUPlot diagram:



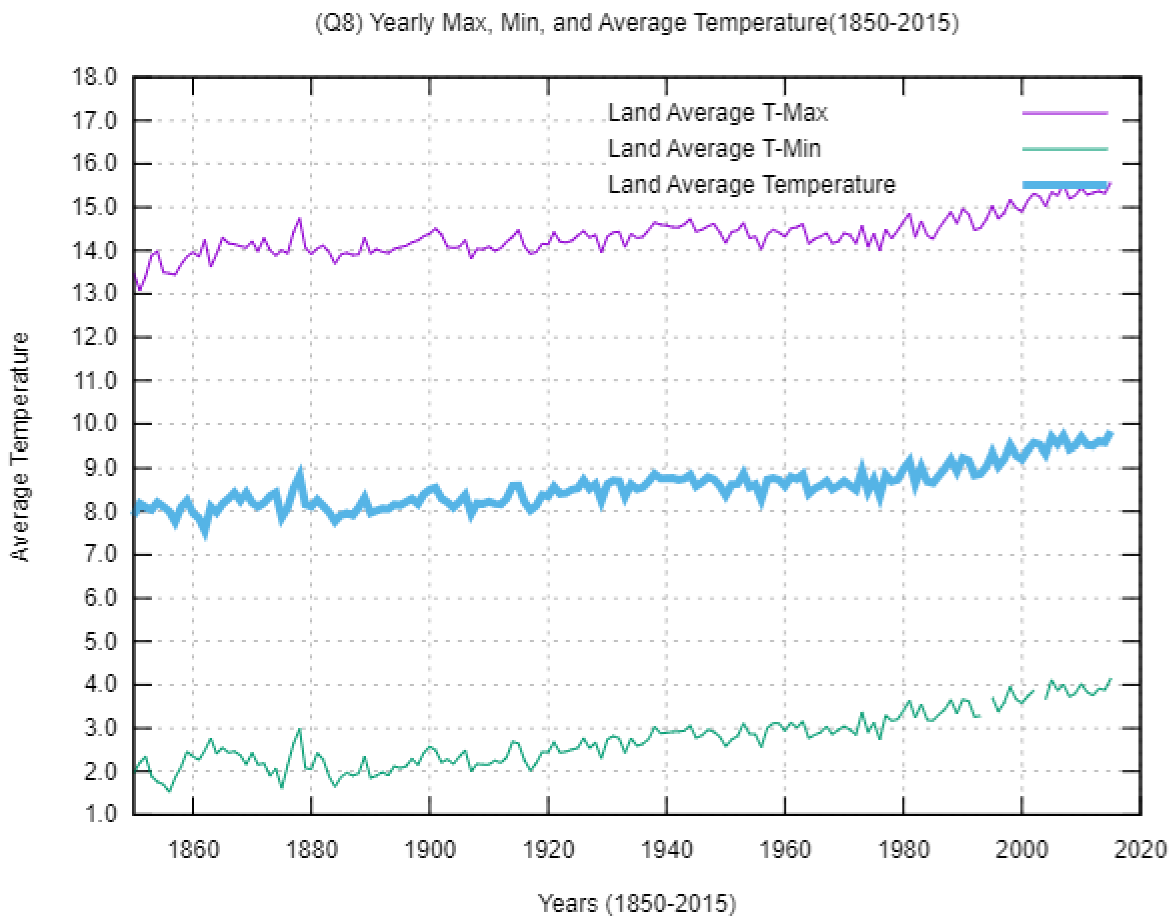
The graph displays average land temperatures for the 19th and 20th centuries, with pink representing the 19th century and green the 20th. Notably, at approximately 7, 25, and 76 years, both lines intersect, indicating similar average temperatures around 8.4, 8.6, and 8.6 degrees, respectively. This convergence allows for comparison between historical climate patterns, helping in understanding temperature trends over the two centuries.

Part 8:

Using the columns LandAverageTemperature, LandMaxTemperature and LandMinTemperature, generate a GNUPlot data file and use GNUPlot to make line plots that show all three temperatures on the same figure. Use the years for the x-axis (use only the years between 1850 and 2015) and the yearly averages for the y-axis. Use three different colors (or different line styles) for the three lines. Make sure that the line plotting LandAverageTemperature stands out from the other two (ex: make that line thicker). Make sure your graph has a title, axes labels and a legend that explicitly tells which line is which.

In creating a line plot for yearly land average, maximum, and minimum temperatures from 1850 to 2015 using GNUPlot, we first prepare a data file containing these temperatures. Then, in GNUPlot, we plot each temperature type on the same figure using different colors or line styles to differentiate them. To emphasize the Land Average Temperature, we ensure its line is thicker than the others. We label the axes clearly, add a descriptive title, and include a legend indicating the represented temperatures. Finally, we display the plot.

GNUPlot diagram:



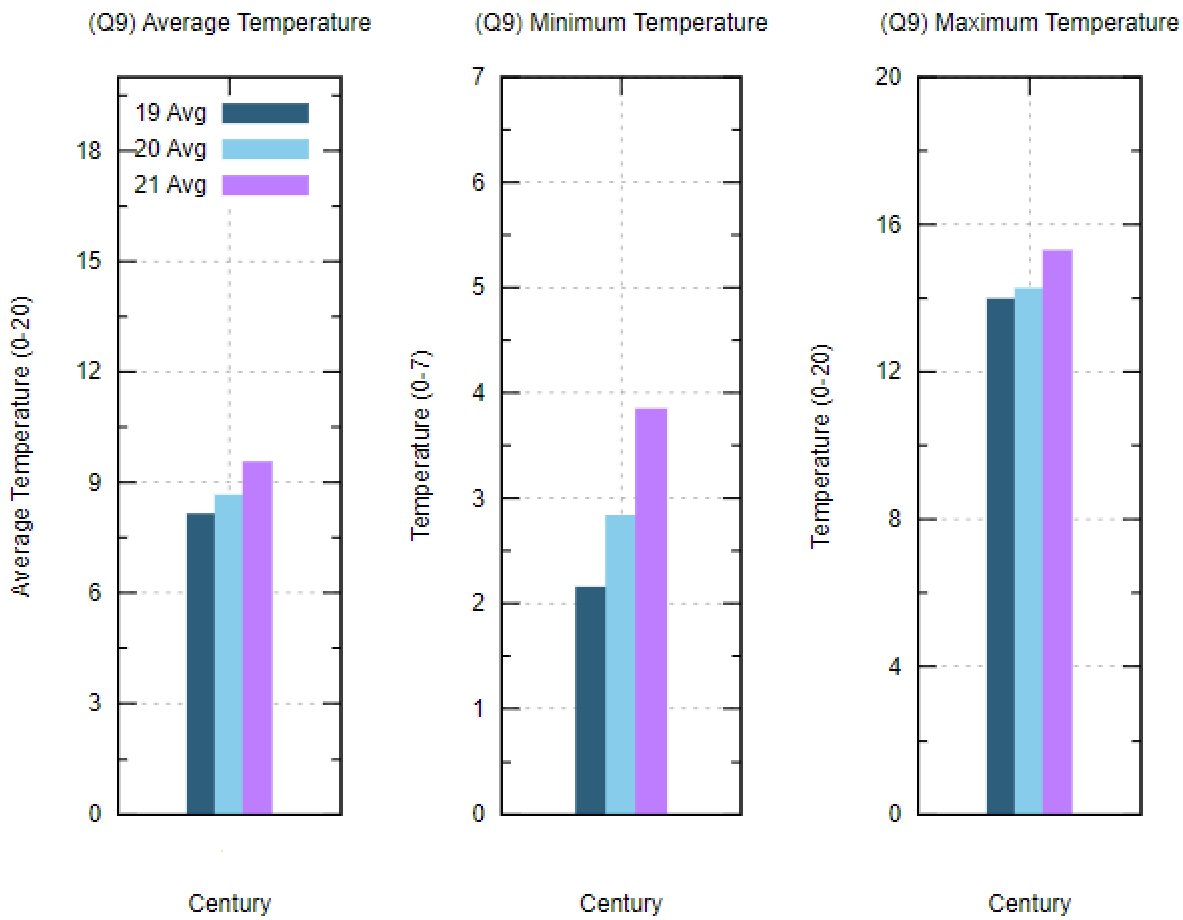
The GNUPlot output displays yearly land temperatures (1850-2015) with three lines representing land average, max, and min temperatures. The Land Average Temperature line, highlighted in blue, is thicker for emphasis, while max and min lines are in pink and green, aiding distinction. Despite color differentiation, the plot shows minimal fluctuation, indicating consistent temperature trends across all metrics. This stability underscores the reliability of the climatic data presented.

Part 9:

Using the columns LandAverageTemperature, LandMaxTemperature and LandMinTemperature, generate a GNUPlot data file and use GNUPlot to make 3 bar plots (box or histogram plots) that show the average, low and high temperatures for each of the 19th (after 1850), 20th and 21st centuries. Put all 3 plots on the same figure displayed as subplots (multiplots). Each plot will show the boxes with different colors (one color per century). Have a title to your figure and have the low or average or high as the title on each subplot.

To generate three bar plots (box or histogram plots) representing the average, low, and high temperatures for each of the 19th, 20th, and 21st centuries as subplots using GNUPlot, we first create separate data files for each century's temperature data. Then, in GNUPlot, we arrange these plots as subplots within a single figure, assigning different colors to distinguish between the centuries. We add titles to each subplot indicating the temperature type and a title to the entire figure for context. After labeling the axes clearly, we display the figure.

GNUPlot diagram:



In the output graph generated by Gnuplot, three bar plots are displayed as subplots within a single figure. Each subplot represents the average, low, and high temperatures for the 19th, 20th, and 21st centuries, respectively. To distinguish between the centuries, unique colors are assigned to each plot. In the three bar plots, all temperature types reached their lowest values during the 19th century, depicted in dark blue, and peaked in the 21st century, represented by pink.

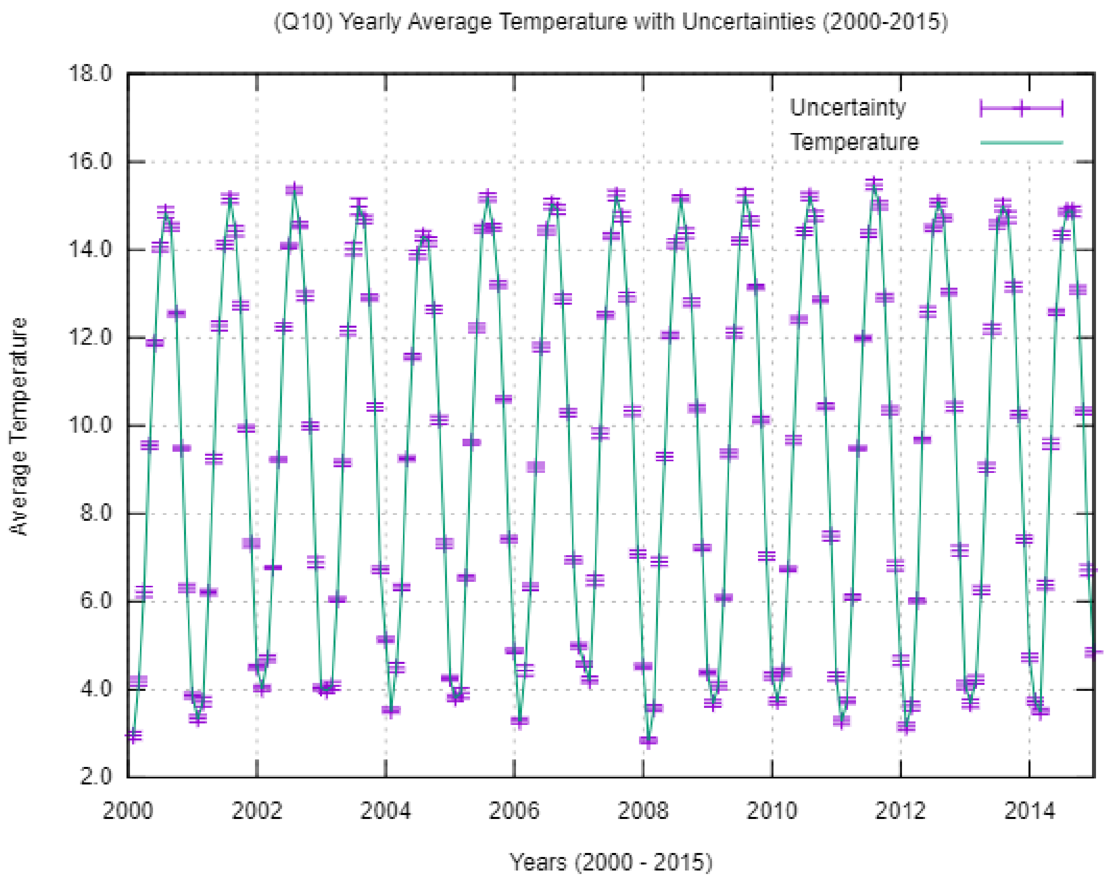
This visualization allows for easy comparison of temperature trends across different centuries.

Part 10:

For the years 2000 to 2015, generate a GNUPlot data file and use GNUPlot to make an error bar plot of the average land temperature by month. Use the uncertainty column for land temperatures to draw the error bars.

For creating an error bar plot of the average land temperature by month with uncertainty for the years 2000 to 2015 using GNUPlot, we generate a data file containing this information. With GNUPlot, we plot the data, including error bars using the uncertainty column. We label the axes clearly and add a descriptive title to the graph. Once plotted, we display the error bar plot for visualization.

GNUPlot diagram:



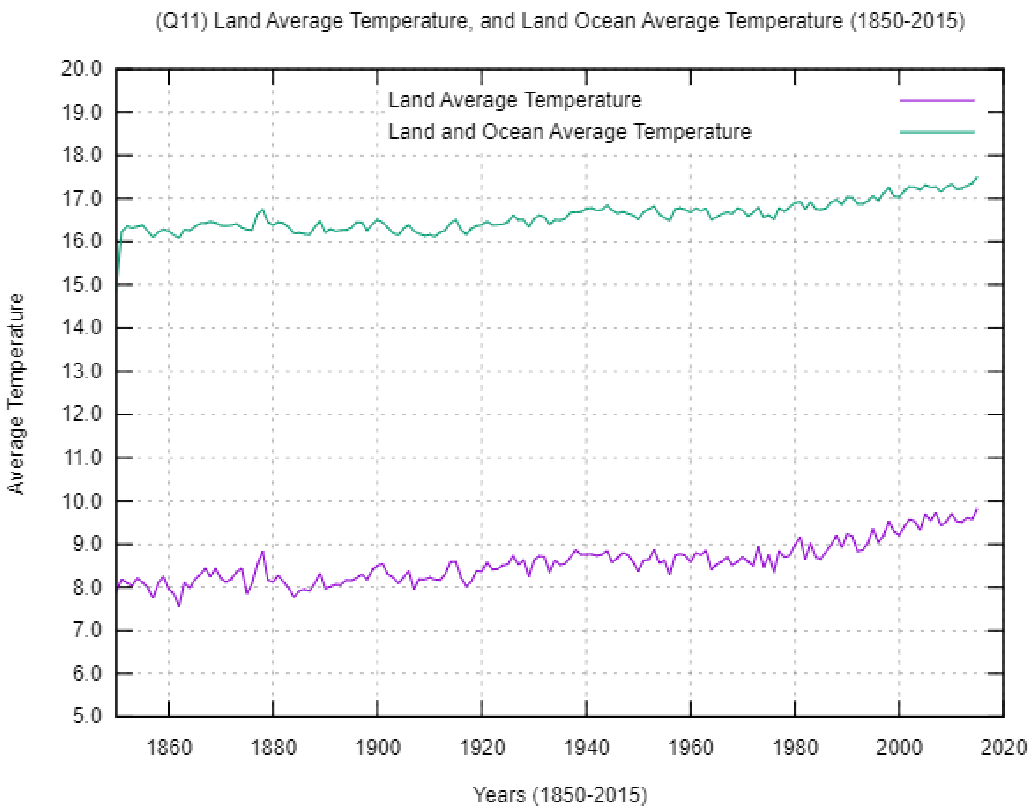
The graph effectively displays the monthly fluctuations in average land temperature and their respective uncertainties, helping in the interpretation of temperature trends over the specified period. In the GNUPlot output, the error bar plot illustrates the average land temperature by month from 2000 to 2015, with average temperatures depicted by a green line and uncertainties indicated by pink error bars. The visualization highlights temperature variations resembling a wave-like pattern, showing the fluctuations of the average temperatures throughout the years, and their uncertainties.

Part 11:

Generate a GNUPlot data file and use GNUPlot to do a plot similar to what you did in question 6 (*LandAverageTemperature*) but only for the years 1850 to 2015 and add the data for the *LandAndOceanAverage Temperatures* column. Have the two lines on the same figure. Label the axes clearly and add a title and legend to your graph.

To plot yearly land average temperatures from 1850 to 2015 alongside Land and Ocean Average Temperatures using GNUPlot, we prepare a data file containing both sets of temperatures. Then, in GNUPlot, we create a line plot for each temperature type on the same figure. We ensure the axes are labeled clearly, add a descriptive title, and include a legend indicating the represented temperatures. After generating the plot, we display it for visualization.

GNUPlot diagram:



The output graph depicts the yearly land average temperatures and combined land/ocean average temperatures from 1850 to 2015, showcasing mostly consistent trends with minimal fluctuations. The pink line representing land average temperatures and the green line representing land/ocean average temperatures exhibit a stable pattern over the specified period, with slight variations observed. This consistency suggests a strong correlation between land and ocean temperatures, contributing to the overall stability of the temperature trends depicted in the plot.

Conclusion:

The completion of this project has strengthened our understanding of C programming and GNUPlot scripting, providing valuable scientific insights into climate trends and patterns through the analysis of real-world temperature data spanning centuries. The visual representations generated by GNUPlot have effectively highlighted notable trends such as the significant rise in average temperatures, particularly evident in the 21st century; aligning with the concept of global warming. The identification of the hottest and coldest months and years further solidifies our understanding of the historical climate fluctuations, which is crucial for informing future climate modeling and adaptation strategies.

Reflecting on the project, effective communication and time management were crucial in overcoming challenges and ensuring timely completion. While each team member contributed their strengths to incorporate a collaborative environment, challenges such as managing large volumes of data and compiler issues underscore the need for more efficient programming techniques. Such large volumes of data resulted in integer overflows occurring especially depending on which compiler was used to read the data file and the program. This would result in unexpected and unreasonable numbers being calculated that would also change every time the code was rerun. Meanwhile, compilers such as OnlineGBD were completely unable to handle the data file due to its large size.

Exploring advanced concepts like dynamic memory allocation and optimized algorithms, along with personal or user-defined libraries for modularization and code reuse, presents opportunities for optimizing memory usage and enhancing program efficiency. Moving forward, we intend to adopt a more unified approach in future projects, where each team member works simultaneously on each section of the project to allow for a smoother workflow. Additionally, this approach also allows each member to gain a deeper understanding of the entire scope of the project.