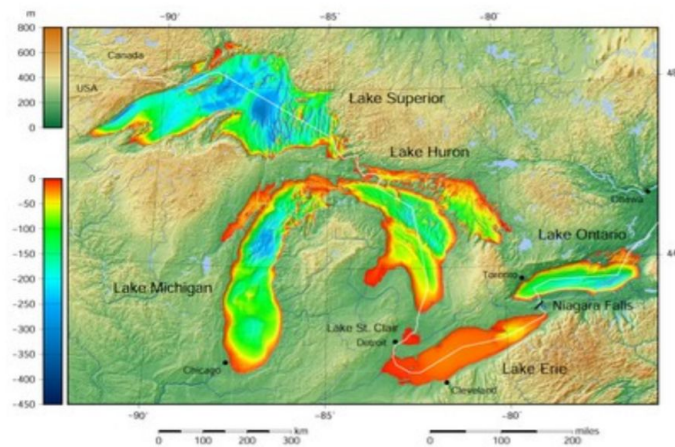


Term Project
Ryerson University
CPS125
Winter 2020
Instructor: Dr. Cherie Ding
April 13, 2020



Tahmeena Mohammad Hashim (500786645), Section: 11, Elements: 1, 2 & 9
Dennon Carafa (500975880), Section: 11, Elements: 3 & 4
Reza Aablue(500966944), Section: 10, Elements: 5 & 6
Aram Ebadi Fard Azar(500963606), Section: 11, Elements: 7 & 8

Introduction:

The purpose of this project is to perform a variety of calculations and conclusions based on the data collected by the National Oceanic and Atmospheric Administration. The data sets consist of 2019's and 2018's daily average surface temperatures for the following six lakes: Lake Ontario, Lake Erie, Lake Michigan, Lake Huron, Lake Superior and Lake St. Clair. The first two parts of the calculations include the yearly averages for each lake in 2019, followed by determining which lake was the warmest and which lake was the coldest. In addition, the C code for parts 1 and 2 also determined which lakes were above the average for all six lakes combined and which lakes fell below that average. Then, further calculations and analysis were made, such as the warmest and coldest days for each lake. Furthermore, for each lake, the summer and winter average temperatures were established. Also, the number of days that it was comfortable to swim in any of the lakes was determined. Those results were followed by the number of the days that each lake was frozen. In the final part of this project, part 1 was repeated but with the 2018's data set. The possible intended audience of this report could be companies who do studies on the great six lakes.

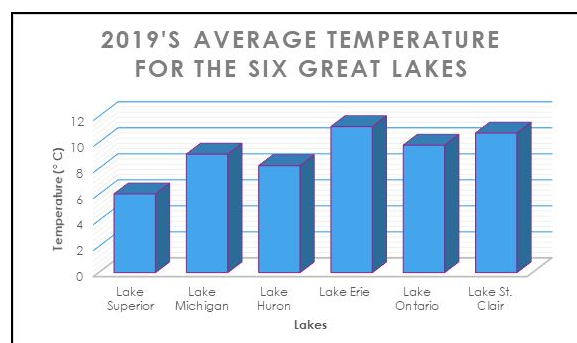
Part1:

Table 1: Each lake's average temperature for the year 2019

Lake Name	Yearly Average Temperature (° C)
Lake Superior	6.05
Lake Michigan	9.11
Lake Huron	8.22
Lake Erie	11.23
Lake Ontario	9.80
Lake St. Clair	10.73

The yearly average temperature for all the six lakes combined is 9.19 (° C)

Graph 1:



By looking at graph 1, the yearly average temperature of the great six lakes is very similar overall. No drastic difference can be observed. This could be due to the location of the great lakes. They are located around similar geographical areas. Also, 9.19 (° C) which is the yearly average temperature for all the six lakes combined, makes sense because both Lake Michigan and Lake Ontario have the yearly average temperature around 9 (° C). Therefore, generally the average tends to be closer to the entities which are repeated the most.

Code output:

```
Element 1
-> Below is the yearly average temperature (degrees C) for each of the lakes:

| Lake | Temperature |
|-----|-----|
| Superior | 6.05 |
| Michigan | 9.11 |
| Huron | 8.22 |
| Erie | 11.23 |
| Ontario | 9.80 |
| St.Clair | 10.73 |
|-----|-----|

-> Yearly average temperature for all the six lakes: 9.19 (degrees C).
-----
```

Part2:

```
Element 2
The coldest lake: Superior
The coldest temperature is 6.05
The warmest lake: Erie
The warmest temperature is 11.23
-----

Lakes with temperatures greater than the average temperature for all the six lakes are:
Erie
Ontario
St. Clair

Lakes with temperatures lower than the average temperature for all the six lakes are:
Superior
Michigan
Huron
-----
(program exited with code: 0)
Press any key to continue . . .
```

By observing the data from table 1 and graph 1 it can be concluded that Superior is the coldest lake with yearly average temperature of 6.05 (° C) while Erie is the warmest lake with yearly average temperature of 11.23 (° C). This makes sense because Lake Superior is closest to the northern hemisphere so it would have the coldest temperature while Lake Erie is mostly located in the south east of Canada's map thus it would have the warmest temperature.

Part3:

Table 3.1: Highest temperature for each lake in 2019 and the day it occurred

Lake Name	Highest Temperature in 2019 (° C)	Date
Lake Superior	16.60	August 17
Lake Michigan	21.75	August 20
Lake Huron	20.41	August 7
Lake Erie	24.86	August 6
Lake Ontario	23.18	August 6
Lake St. Clair	25.04	July 20

Graph 3.1:

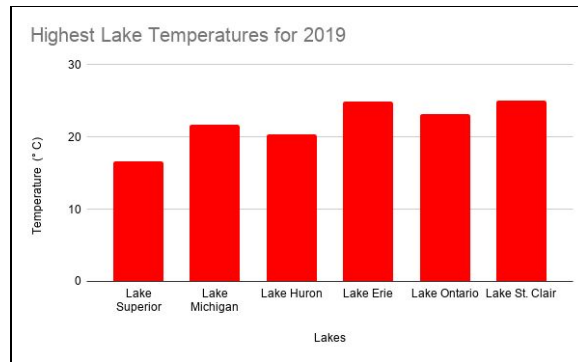
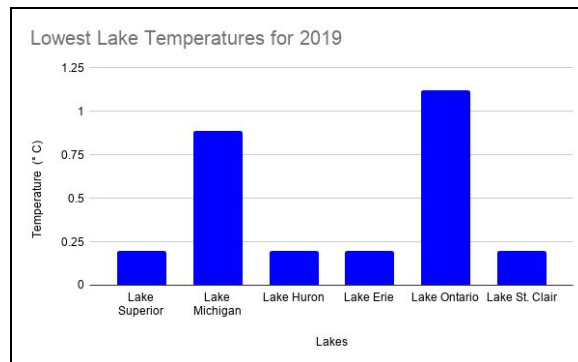


Table 3.2: Lowest temperature for each lake in 2019 and the day it occurred

Lake Name	Lowest Temperature in 2019 (° C)	Date
Lake Superior	0.20	March 5
Lake Michigan	0.89	March 8
Lake Huron	0.20	March 1
Lake Erie	0.20	February 1
Lake Ontario	1.12	March 1
Lake St. Clair	0.20	January 19

Graph 3.2:



From Table 3.1, it is evident that the warmest temperatures for the lakes occur during the summer, around late July and mid-August. From Table 3.2, the coldest temperatures occur during winter, from late January to early March. The warmest temperatures of the lakes vary from 16.60° C to 25.04° C. The coldest temperatures of the lakes vary from 0.20° C to 1.12° C.

Code Output:

```

Warmest Lake Temperatures for 2019
-----
Lake Superior : August 17, 16.60 degrees Celsius
Lake Michigan : August 20, 21.75 degrees Celsius
Lake Huron    : August 7, 20.41 degrees Celsius
Lake Erie     : August 6, 24.86 degrees Celsius
Lake Ontario  : August 6, 23.18 degrees Celsius
Lake St. Clair: July 20, 25.04 degrees Celsius

Coldest Lake Temperatures for 2019
-----
Lake Superior : March 5, 0.20 degrees Celsius
Lake Michigan : March 8, 0.89 degrees Celsius
Lake Huron    : March 1, 0.20 degrees Celsius
Lake Erie     : February 1, 0.20 degrees Celsius
Lake Ontario  : March 1, 1.12 degrees Celsius
Lake St. Clair: January 19, 0.20 degrees Celsius

-----
(program exited with code: 0)
Press any key to continue . . .

```

Part4:

```

Warmest Overall Temperature of All the Lakes for 2019
-----
Lake St. Clair: July 20, 25.04 degrees Celsius

Coldest Overall Temperature of All the Lakes for 2019
-----
Lake St. Clair: January 19, 0.20 degrees Celsius

-----
(program exited with code: 0)
Press any key to continue . . .

```

From looking at Graph 3.1 and Graph 3.2, it is easy to see which lake has the highest temperature and which lake has the lowest temperature. From Graph 3.1, Lake St. Clair has the highest temperature out of all the lakes, with Lake Erie having the second highest. From Graph 3.2, the lakes with the lowest lake temperatures are Lake Superior, Lake Huron, Lake Erie, and Lake St. Clair. The warmest of the lake temperatures is Lake St. Clair, having a temperature of 25.04° C. The coldest temperature is 0.20° C which was at Lake St. Clair.

Part 5:

Table 5: Average Summer Temperatures amongst the 6 lakes (Degrees Celsius/day)

Lake Name	Average Summer Temperature (° C/day)
Lake Erie	22.86
Lake St. Clair	22.43
Lake Ontario	20.57
Lake Michigan	19.05
Lake Huron	17.74
Lake Superior	12.59

Based on the current calculations and observations for the summer season, Lake Erie is the warmest lake and Lake Superior is the coldest lake on average during the summer season. This indicates

that the lakes located closer to the East of Toronto, Ontario, tend to have warmer temperatures on a daily average during the summer. In terms of comparison with results from part 2 of the report, the results are both consistent, where Lake Erie is the warmest lake and Lake Superior is the coldest lake, based on averages for the whole year and the summer season.

Code output for part 5:

```
The temperature averages are listed from warmest to coldest.

Average Summer Temperatures amongst the 6 lakes (Degrees Celsius/day)
-----
1. 22.86 (Lake Erie)
2. 22.43 (Lake St. Clair)
3. 20.57 (Lake Ontario)
4. 19.05 (Lake Michigan)
5. 17.74 (Lake Huron)
6. 12.59 (Lake Superior)
```

Part 6:

Table 6: Average Winter Temperatures amongst the 6 lakes (Degrees Celsius/day)

Lake Name	Average Winter Temperature (° C/day)
Lake Ontario	2.73
Lake Michigan	2.43
Lake Huron	1.56
Lake Superior	1.47
Lake Erie	1.47
Lake St. Clair	0.89

During the winter season, Lake Ontario is the warmest lake and Lake St. Clair is the coldest lake, based on a daily average. This implies that in the winter season, the lakes located closer to East of Toronto, Ontario, tend to be colder than those on Toronto's west side. When these results are compared with the project's results from part 2, the results are not consistent, as Lake Superior and Lake Erie are two of the coldest lakes during the winter season. This is logical according to Geographical reasoning and explanation, as both lakes are, on average, close to freezing point daily temperatures during the winter season.

Code output for part 6:

The temperature averages are listed from warmest to coldest.

Average Winter Temperatures amongst the 6 lakes (Degrees Celsius/day)

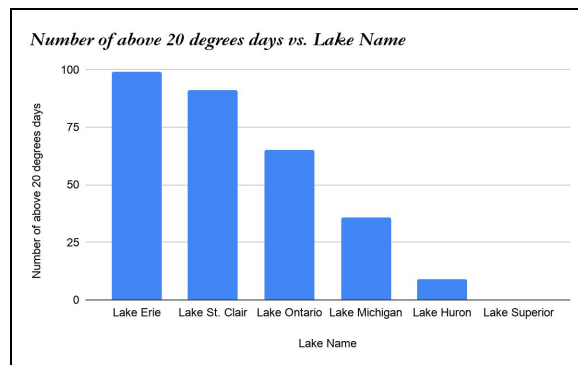
1. 2.73 (Lake Ontario)
2. 2.43 (Lake Michigan)
3. 1.56 (Lake Huron)
4. 1.47 (Lake Superior)
5. 1.47 (Lake Erie)
6. 0.89 (Lake St. Clair)

Part 7:

Table 7: shows the number of days the that temperature is above 20 degrees

Lake Name	Number of above 20 degrees days
Lake Erie	99
Lake St. Clair	91
Lake Ontario	65
Lake Michigan	36
Lake Huron	9
Lake Superior	0

Graph 7:



By observing the following data, Lake Erie has more days for which the temperature is more than 20 degrees because according to the map Lake Erie tends to have a hotter atmosphere. Respectively, Lake St.Clair, Lake Ontario, Lake Michigan, Lake Huron, Lake Superior are the next ones and they are in less red areas and regions.

Part 8:

Table 8: Displaying the number of days for which the temperature was below 0 degrees

Lake Name	Number of frozen days
Lake Erie	0
Lake St. Clair	0
Lake Ontario	0
Lake Michigan	0
Lake Huron	0
Lake Superior	0

The results make sense because most of the lakes tend to have warmer atmospheres and having zero days of temperature below zero degrees is totally logical. In fact, part 7 results are almost all above 20 degrees, so it's logical that all of them have no freezing day.

Code Output for parts 7 and 8 (merged together):

```
The number of days in the 2019 that one could swim comfortably in the great six lakes, assuming that the temperature above 20 degrees was ideal for swimming

Lake Superior
Above 20 degrees: 0, Below 0 degrees: 0
Lake Michigan
Above 20 degrees: 36, Below 0 degrees: 0
Lake Huron
Above 20 degrees: 9, Below 0 degrees: 0
Lake Erie
Above 20 degrees: 99, Below 0 degrees: 0
Lake Ontario
Above 20 degrees: 65, Below 0 degrees: 0
Lake St.Clair
Above 20 degrees: 91, Below 0 degrees: 0

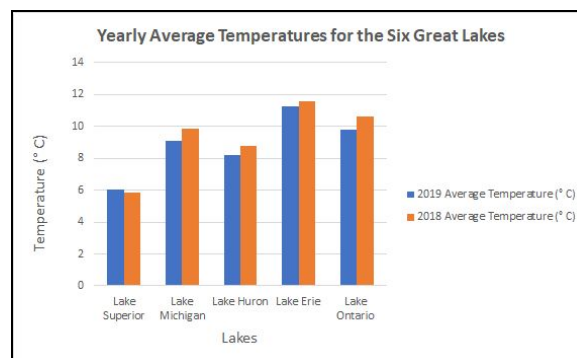
-----
(program exited with code: 0)
Press any key to continue . . .
```

Part9:

Table 9: Each lake's average temperature for the year 2019 and as well as 2018

Lake Name	2019 Average Temperature (° C)	2018 Average Temperature (° C)
Lake Superior	6.05	5.87
Lake Michigan	9.11	9.87
Lake Huron	8.22	8.78
Lake Erie	11.23	11.56
Lake Ontario	9.80	10.60
Lake St. Clair	10.73	11.20

Graph 9:



Overall, the yearly average temperature of all the lakes for 2019 has slightly decreased compared to the year 2018. Excluding Lake Superior all the other yearly average temperatures for the other lakes are less than the yearly average temperature for the year 2019. According to [1] the yearly average temperature of the six great lakes generally stays the same. The difference between the two year's average temperatures might just be the result of normal temperature fluctuation which normally happens from time to time.

Code output:

```
-> Below is the yearly average temperature (degrees C) for each of the lakes:

-----
| 2019 average Temp. | 2018 average Temp. |
|-----|-----|
| Lake Superior : 6.05 | Lake Superior : 5.87 |
| Lake Michigan : 9.11 | Lake Michigan : 9.87 |
| Lake Huron : 8.22 | Lake Huron : 8.78 |
| Lake Erie : 11.23 | Lake Erie : 11.56 |
| Lake Ontario : 9.80 | Lake Ontario : 10.60 |
| Lake St.Clair : 10.73 | Lake St.Clair : 11.20 |
|-----|-----|

-> 2019 average temperature for all the six lakes: 9.19 (degrees C).
-> 2018 average temperature for all the six lakes: 9.65 (degrees C).

-----
(program exited with code: 0)
Press any key to continue . . . █
```

Conclusion:

Overall, completing this project was a very educational experience. Not only did working on this project improve our understanding of the course material, but it also allowed us to experience what to expect while working in a cooperative group project. In addition, one thing that made the project very fun was that we could use our knowledge of C programming and geography at the same time. This allows us to see real-life applications for C programming. One of the challenges was working with two dimensional arrays for the first time. The biggest challenge was making the report to have a consistent style while having the work from four different people.

If we had to re-do this project, we would use helping functions. Helping functions would increase the efficiency of our codes and reduce the number of lines that we had to implement in our code. We would also discuss the format and style of the document prior to the start of the project so that way we would have reduced the time that it took for making all the elements to have a consistent style and format.

Reference:

[1]. Node, CoastWatch Great Lakes. *Great Lakes Statistics*, coastwatch.glerl.noaa.gov/statistic/.

Appendix

Part 1 and Part2 codes (merged together):

```
/* CPS 125 Term Project - Parts 1 and 2 */
/* Name: Tahmeena Mohammad Hashim */
/* Section: 11 */
/* Student Number: 5007866 */

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int main()

{
//Part 1
    system("color f0");
    int i,j;
    double temp[365][8]; //initializing the array
    double
sum_sup=0,avg_sup=0,sum_mich=0,avg_mich=0,sum_huron=0,avg_huron=0,sum_erie=0,avg_erie=0,sum_ont=0,avg_ont=0,sum_st_clr=0,avg_st_clr=0,total_avg;

    FILE*in=fopen("data_2019.txt", "r"); //opening the 2019 data file
    for (i=0;i<365;i++)//For loop to go through rows
    {
        for (j=0;j<8;j++) //For loop to go through columns
        {
            fscanf(in,"%lf",&temp[i][j]); //scanning elements in rows and columns
        }
        sum_sup=sum_sup+temp[i][2]; //Finding the averages for each column of the file
        avg_sup=sum_sup/365;

        sum_mich=sum_mich+temp[i][3];
        avg_mich=sum_mich/365;

        sum_huron=sum_huron+temp[i][4];
        avg_huron=sum_huron/365;

        sum_erie=sum_erie+temp[i][5];
        avg_erie=sum_erie/365;
```

```

sum_ont=sum_ont+temp[i][6];
avg_ont=sum_ont/365;

sum_st_clr=sum_st_clr+temp[i][7];
avg_st_clr=sum_st_clr/365;

}
printf("Element 1\n");
printf(" \n-> Below is the yearly average temperature (degrees C) for each of the lakes:\n\n");
printf(" ----- \n");

printf(" | Lake | Temperature | \n");
printf(" ----- \n");

printf(" | Superior | %.2lf | \n",avg_sup);

printf(" | Michigan | %.2lf | \n",avg_mich);

printf(" | Huron | %.2lf | \n",avg_huron);

printf(" | Erie | %.2lf | \n",avg_erie);

printf(" | Ontario | %.2lf | \n",avg_ont);

printf(" | St.Clair | %.2lf | \n",avg_st_clr);

total_avg = (avg_sup + avg_mich + avg_huron + avg_erie + avg_ont + avg_st_clr)/6;

printf(" ----- \n\n");

printf("-> Yearly average temperature for all the six lakes: %.2lf (degrees C).\n\n",total_avg);

printf("----- \n");

```

//Part 2

```

printf("Element 2\n\n");
double avgs[6] = {avg_sup, avg_mich, avg_huron, avg_erie, avg_ont, avg_st_clr}; // Storing the
averages (determined in part 1) in an array of size 6
double avgs_2[6] = {avg_sup, avg_mich, avg_huron, avg_erie, avg_ont, avg_st_clr};
double coldest, warmest;

warmest = avgs[0];

```

```

coldest = avgs_2[0];

for (i=1;i<6;i++)
{
    if (warmest<avgs[i]) //compares the variable with the first array
    {
        warmest = avgs[i];
    }
    if (coldest>avgs_2[i]) //compares the variable with the second array
    {
        coldest = avgs_2[i];
    }
}

char* name[6]={"Superior","Michigan","Huron","Erie","Ontario","St. Clair"}; //Using pointer
array to print the name of the warmest lake and as well as coldest
int num;

for (num = 0; num <6; num++)
{
    if (warmest == avgs[num])
    {
        printf ("\nThe warmest lake:          ");

        puts(name[num]);
        printf("The warmest temperature is      %.2lf\n",warmest);

    }

    if (coldest == avgs_2[num])
    {
        printf ("The coldest lake:          ");
        puts(name[num]);
        printf("The coldest temperature is      %.2lf", coldest);

    }
}

printf("\n-----\n");

printf("\nLakes with temperatures greater than the average temperature for all the six lakes are:
\n");

int great;
for (great = 0; great < 6; great++) //Using for loop to compare the total average with the averages

```

```

for each lake
{
    if ( avgs[great] > total_avg)
    {
        puts(name[great]);
    }
}

printf("\nLakes with temperatures lower than the average temperature for all the six lakes are:
\n");

int low;
for (low = 0; low < 6; low++)
{
    if ( avgs[low] < total_avg)
    {
        puts(name[low]);
    }
}

fclose(in);
return (0);
}

```

Part 3 code:

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h> //This library includes the string function needed for copying strings to variables//
/* CPS125 Term Project Part 3
 *
 * Name: Dennon Carafa
 * Student Number: 500975880
 */
/* This function takes one of the arrays of the lake temperatures and
 * determines the highest temperature out of all the recorded temperatures for
 * that lake that was read from the file. It also uses a pointer variable
 * to record the day the highest temperature took place.
 */

double highest_temp(const double temp[], int size, int *day){
    int day_of_high_temp;
    double highest_temp;
    highest_temp = temp[0];

```

```

/* This loop checks each of the recorded lake temperatures and
 * compares them to the current recorded highest temperature named
 * highest_temp. If the current highest temperature is less than the
 * temperature for that day, that day's temperature will be placed in
 * the variable highest_temp.
 */
for (int i = 1; i <= size; i++){

    if (highest_temp < temp[i-1]){
        highest_temp = temp[i-1];
        day_of_high_temp = i;
    }
}

*day = day_of_high_temp; //Returns day of the highest temperature//

return(highest_temp); //Returns highest temperature back to main function//
}

/* This function takes one of the arrays of the lake temperatures and
 * determines the lowest temperature out of all the recorded temperatures for
 * that lake that was read from the file. It also uses a pointer variable
 * to record the day the lowest temperature took place.
 */
double lowest_temp(const double temp[], int size, int *day){
    int day_of_low_temp;
    double lowest_temp;
    lowest_temp = temp[0];

    /* This loop checks each recorded temperature for the lake and will
     * compare them to the current lowest temperature.
     */
    for (int i = 1; i <= size; i++){

        if (lowest_temp > temp[i-1]){
            lowest_temp = temp[i-1];
            day_of_low_temp = i;
        }
    }

    *day = day_of_low_temp; //Returns day of the lowest temperature//

    return(lowest_temp); //Returns the lowest temperature of the lake//
}

```

```

}
int date(int day_in_year, char lake_month[10]){

/* This function takes the day from the highest_temp or lowest_temp
 * functions and converts that day into the standard day/month format.
 * This function only considers a 365 day year.
 */
int day;
char months[12][10] = {"January", "February", "March", "April", "May", "June", "July", "August",
"September", "October", "November", "December"};
if (day_in_year >= 1 && day_in_year <= 31){
    day = day_in_year;
    strcpy (lake_month, months[0]);
} else
if (day_in_year >= 32 && day_in_year <= 59){
    day = day_in_year - 31;
    strcpy (lake_month, months[1]);
} else
if (day_in_year >= 60 && day_in_year <= 90){
    day = day_in_year - 59;
    strcpy (lake_month, months[2]);
} else
if (day_in_year >= 91 && day_in_year <= 120){
    day = day_in_year - 90;
    strcpy (lake_month, months[3]);
} else
if (day_in_year >= 121 && day_in_year <= 151){
    day = day_in_year - 120;
    strcpy (lake_month, months[4]);
} else
if (day_in_year >= 152 && day_in_year <= 181){
    day = day_in_year - 151;
    strcpy (lake_month, months[5]);
} else
if (day_in_year >= 182 && day_in_year <= 212){
    day = day_in_year - 181;
    strcpy (lake_month, months[6]);
} else
if (day_in_year >= 213 && day_in_year <= 243){
    day = day_in_year - 212;
    strcpy (lake_month, months[7]);
} else
if (day_in_year >= 244 && day_in_year <= 273){

```

```

    day = day_in_year - 243;
    strcpy (lake_month, months[8]);
} else
if (day_in_year >= 274 && day_in_year <= 304){
    day = day_in_year - 273;
    strcpy (lake_month, months[9]);
} else
if (day_in_year >= 305 && day_in_year <= 334){
    day = day_in_year - 304;
    strcpy (lake_month, months[10]);
} else
if (day_in_year >= 335 && day_in_year <= 365){
    day = day_in_year - 334;
    strcpy (lake_month, months[11]);
}

return(day);
}
int main(void)
{

    system ("color f0");
    //All declarations for the days, months, temperatures of the lakes//
    int i, day_high[6], day_low[6], day_of_year, year;
    char month[10], month_high[6][10], month_low[6][10], names_of_lakes[6][10] = {"Superior",
    "Michigan", "Huron", "Erie", "Ontario", "St. Clair"};
    double sup_temp[365], mich_temp[365], huron_temp[365], erie_temp[365], ont_temp[365],
    stclr_temp[365], highest_lake_temp[6], lowest_lake_temp[6];

    char *filename = "2019LakeTemperatures.txt";

    //Opens the file//
    FILE *in;
    in = fopen("2019LakeTemperatures.txt", "r");

    //Checks status of file//
    if(in == NULL){

        printf("There was an error opening the file %s \n", filename);
        return(1);
    }

    //Reads the data from the file and puts it into its respective lake temperature array//

```



```

        while (!feof(in)){

fscanf(in, "%d", &year);
fscanf(in, "%d", &i);
fscanf(in, "%lf", &sup_temp[i-1]);
fscanf(in, "%lf", &mich_temp[i-1]);
fscanf(in, "%lf", &huron_temp[i-1]);
fscanf(in, "%lf", &erie_temp[i-1]);
fscanf(in, "%lf", &ont_temp[i-1]);
fscanf(in, "%lf", &stclr_temp[i-1]);

}

fclose(in); //Closes the file//

/* These function calls send the array for each lakes temperature as well
* as their size to the highest_temp function and lowest_temp function to find
* the highest and lowest temperatures of each Lake. The third argument is the
* address for a variable that stores the date of the highest or lowest temperature
* in the main function. The date is then passed to the date function converts the
* date into a familiar day/month format that can be easily understood by the user.
*/
highest_lake_temp[0] = highest_temp(sup_temp, 365, &day_of_year);
day_high[0] = date(day_of_year, month);
strcpy (month_high[0], month); //Copies the name of the month into an array of the months of the highest
temperatures//
lowest_lake_temp[0] = lowest_temp(sup_temp, 365, &day_of_year);
day_low[0] = date(day_of_year, month);
strcpy (month_low[0], month);

highest_lake_temp[1] = highest_temp(mich_temp, 365, &day_of_year);
day_high[1] = date(day_of_year, month);
strcpy (month_high[1], month);
lowest_lake_temp[1] = lowest_temp(mich_temp, 365, &day_of_year);
day_low[1] = date(day_of_year, month);
strcpy (month_low[1], month);

highest_lake_temp[2] = highest_temp(huron_temp, 365, &day_of_year);
day_high[2] = date(day_of_year, month);
strcpy (month_high[2], month);
lowest_lake_temp[2] = lowest_temp(huron_temp, 365, &day_of_year);
day_low[2] = date(day_of_year, month);

```

```

strcpy (month_low[2], month);

highest_lake_temp[3] = highest_temp(erie_temp, 365, &day_of_year);
day_high[3] = date(day_of_year, month);
strcpy (month_high[3], month);
lowest_lake_temp[3] = lowest_temp(erie_temp, 365, &day_of_year);
day_low[3] = date(day_of_year, month);
strcpy (month_low[3], month);

highest_lake_temp[4] = highest_temp(ont_temp, 365, &day_of_year);
day_high[4] = date(day_of_year, month);
strcpy (month_high[4], month);
lowest_lake_temp[4] = lowest_temp(ont_temp, 365, &day_of_year);
day_low[4] = date(day_of_year, month);
strcpy (month_low[4], month);

highest_lake_temp[5] = highest_temp(stclr_temp, 365, &day_of_year);
day_high[5] = date(day_of_year, month);
strcpy (month_high[5], month);
lowest_lake_temp[5] = lowest_temp(stclr_temp, 365, &day_of_year);
day_low[5] = date(day_of_year, month);
strcpy (month_low[5], month);

printf("%48s%d\n", "Warmest Lake Temperatures for ", year);
printf("-----\n");
for (i = 0; i < 6; i++){
    printf("Lake %-9s: %s %d, %.2lf degrees Celsius\n", names_of_lakes[i], month_high[i], day_high[i],
highest_lake_temp[i]);
}
printf("\n\n");

printf("%48s%d\n", "Coldest Lake Temperatures for ", year);
printf("-----\n");
for (i = 0; i < 6; i++){
    printf("Lake %-9s: %s %d, %.2lf degrees Celsius\n", names_of_lakes[i], month_low[i], day_low[i],
lowest_lake_temp[i]);
}

return 0;
}

```

Part 4 code:

```
#include <stdio.h>
```

```

#include <stdlib.h>
#include <string.h> //This library includes the string function needed for copying strings to variables//
/* CPS125 Term Project Part 3
*
* Name: Dennon Carafa
* Student Number: 500975880
*/
/* This function takes all of the arrays of the lake temperatures and
* determines the highest temperature out of all the recorded temperatures
* read from the file. It also uses a pointer variable to record the day the
* highest temperature took place.
*/
double highest_temp(const double suptemp[], const double michtemp[], const double hurontemp[], const
double erietemp[], const double onttemp[], const double stclrtemp[], int size, int *day, const char
names_of_lakes[6][10],char lake[10]){
    int day_of_high_temp;
    double highest_temp;
    highest_temp = suptemp[0];

    /* This loop will read all the temperatures from each array of temperatures
    * and determine the highest temperature from all the lakes.
    */
    for (int i = 1; i <= size; i++){

        if (highest_temp < suptemp[i-1] && suptemp[i-1] > michtemp[i-1] && suptemp[i-1] > hurontemp[i-1]
&& suptemp[i-1] > erietemp[i-1] && suptemp[i-1] > onttemp[i-1] && suptemp[i-1] > stclrtemp[i-1]){
            highest_temp = suptemp[i-1];
            day_of_high_temp = i;
            strcpy (lake, names_of_lakes[0]);
        } else
            if (highest_temp < michtemp[i-1] && michtemp[i-1] > suptemp[i-1] && michtemp[i-1] >
hurontemp[i-1] && michtemp[i-1] > erietemp[i-1] && michtemp[i-1] > onttemp[i-1] && michtemp[i-1]
> stclrtemp[i-1]){
                highest_temp = michtemp[i-1];
                day_of_high_temp = i;
                strcpy (lake, names_of_lakes[1]);
            } else
                if (highest_temp < hurontemp[i-1] && hurontemp[i-1] > suptemp[i-1] && hurontemp[i-1] >
michtemp[i-1] && hurontemp[i-1] > erietemp[i-1] && hurontemp[i-1] > onttemp[i-1] &&
hurontemp[i-1] > stclrtemp[i-1]){
                    highest_temp = hurontemp[i-1];
                    day_of_high_temp = i;
                    strcpy (lake, names_of_lakes[2]);
                }
    }
}

```

```

    } else
    if (highest_temp < erietemp[i-1] && erietemp[i-1] > suptemp[i-1] && erietemp[i-1] > michtemp[i-1]
    && erietemp[i-1] > hurontemp[i-1] && erietemp[i-1] > onttemp[i-1] && erietemp[i-1] > stclrttemp[i-1]){
        highest_temp = erietemp[i-1];
        day_of_high_temp = i;
        strcpy (lake, names_of_lakes[3]);
    } else
    if (highest_temp < onttemp[i-1] && onttemp[i-1] > suptemp[i-1] && onttemp[i-1] > michtemp[i-1] &&
    onttemp[i-1] > hurontemp[i-1] && onttemp[i-1] > erietemp[i-1] && onttemp[i-1] > stclrttemp[i-1]){
        highest_temp = onttemp[i-1];
        day_of_high_temp = i;
        strcpy (lake, names_of_lakes[4]);
    } else
    if (highest_temp < stclrttemp[i-1] && stclrttemp[i-1] > suptemp[i-1] && stclrttemp[i-1] > michtemp[i-1]
    && stclrttemp[i-1] > hurontemp[i-1] && stclrttemp[i-1] > erietemp[i-1] && stclrttemp[i-1] >
    onttemp[i-1]){
        highest_temp = stclrttemp[i-1];
        day_of_high_temp = i;
        strcpy (lake, names_of_lakes[5]);
    }

}

*day = day_of_high_temp; //Returns the date of the highest temperature//

return(highest_temp); //Returns the highest temperature of all the lakes//
}
/* This function takes all of the arrays of the lake temperatures and
* determines the lowest temperature out of all the recorded temperatures
* read from the file. It also uses a pointer variable to record the day the
* lowest temperature took place.
*/
double lowest_temp(const double suptemp[], const double michtemp[], const double hurontemp[], const
double erietemp[], const double onttemp[], const double stclrttemp[], int size, int *day, const char
names_of_lakes[6][10], char lake[10]){
    int day_of_low_temp;
    double lowest_temp;
    lowest_temp = suptemp[0];

/* This loop will read all the temperatures from each array of temperatures
* and determine the lowest temperature from all the lakes.
*/
for (int i = 1; i <= size; i++){

```

```

if (lowest_temp > suptemp[i-1]){
    lowest_temp = suptemp[i-1];
    day_of_low_temp = i;
    strcpy (lake, names_of_lakes[0]);
} else
if (lowest_temp > michtemp[i-1]){
    lowest_temp = michtemp[i-1];
    day_of_low_temp = i;
    strcpy (lake, names_of_lakes[1]);
} else
if (lowest_temp > hurontemp[i-1]){
    lowest_temp = hurontemp[i-1];
    day_of_low_temp = i;
    strcpy (lake, names_of_lakes[2]);
} else
if (lowest_temp > erietemp[i-1]){
    lowest_temp = erietemp[i-1];
    day_of_low_temp = i;
    strcpy (lake, names_of_lakes[3]);
} else
if (lowest_temp > onttemp[i-1]){
    lowest_temp = onttemp[i-1];
    day_of_low_temp = i;
    strcpy (lake, names_of_lakes[4]);
} else
if (lowest_temp > stclrtmp[i-1]){
    lowest_temp = stclrtmp[i-1];
    day_of_low_temp = i;
    strcpy (lake, names_of_lakes[5]);
}
}

*day = day_of_low_temp; //Returns the date of the lowest temperature//

return(lowest_temp); //Returns the lowest temperature of all the lakes//
}

int date(int day_in_year, char lake_month[10]){

    /* This function takes the day from the highest_temp or lowest_temp
    * functions and converts that day into the standard day/month format.
    * This function only considers a 365 day year.

```

*/

```
int day;
char months[12][10] = {"January", "February", "March", "April", "May", "June", "July", "August",
"September", "October", "November", "December"};
if (day_in_year >= 1 && day_in_year <= 31){
    day = day_in_year;
    strcpy (lake_month, months[0]);
} else
if (day_in_year >= 32 && day_in_year <= 59){
    day = day_in_year - 31;
    strcpy (lake_month, months[1]);
} else
if (day_in_year >= 60 && day_in_year <= 90){
    day = day_in_year - 59;
    strcpy (lake_month, months[2]);
} else
if (day_in_year >= 91 && day_in_year <= 120){
    day = day_in_year - 90;
    strcpy (lake_month, months[3]);
} else
if (day_in_year >= 121 && day_in_year <= 151){
    day = day_in_year - 120;
    strcpy (lake_month, months[4]);
} else
if (day_in_year >= 152 && day_in_year <= 181){
    day = day_in_year - 151;
    strcpy (lake_month, months[5]);
} else
if (day_in_year >= 182 && day_in_year <= 212){
    day = day_in_year - 181;
    strcpy (lake_month, months[6]);
} else
if (day_in_year >= 213 && day_in_year <= 243){
    day = day_in_year - 212;
    strcpy (lake_month, months[7]);
} else
if (day_in_year >= 244 && day_in_year <= 273){
    day = day_in_year - 243;
    strcpy (lake_month, months[8]);
} else
if (day_in_year >= 274 && day_in_year <= 304){
    day = day_in_year - 273;
```

```

    strcpy (lake_month, months[9]);
} else
if (day_in_year >= 305 && day_in_year <= 334){
    day = day_in_year - 304;
    strcpy (lake_month, months[10]);
} else
if (day_in_year >= 335 && day_in_year <= 365){
    day = day_in_year - 334;
    strcpy (lake_month, months[11]);
}
return(day);
}
int main(void)
{

system("color f0");
//Declarations for the day, month, temperatures//
int i, year, day_of_year, warmest_day, coldest_day;
double sup_temp[365], mich_temp[365], huron_temp[365], erie_temp[365], ont_temp[365],
stclr_temp[365], warmest_temp, coldest_temp;
char lake[10], lake_warm[10], lake_cold[10], month[10], month_high[10], month_low[10],
names_of_lakes[6][10] = {"Superior", "Michigan", "Huron", "Erie", "Ontario", "St. Clair"};

char *filename = "2019LakeTemperatures.txt";

FILE *in;
in = fopen("2019LakeTemperatures.txt", "r");

if(in == NULL){

    printf("There was an error opening the file %s \n", filename);
    return(1);
}

while (!feof(in)){

fscanf(in, "%d", &year);
fscanf(in, "%d", &i);
fscanf(in, "%lf", &sup_temp[i-1]);
fscanf(in, "%lf", &mich_temp[i-1]);
fscanf(in, "%lf", &huron_temp[i-1]);
fscanf(in, "%lf", &erie_temp[i-1]);
fscanf(in, "%lf", &ont_temp[i-1]);

```

```

    fscanf(in, "%lf", &stclr_temp[i-1]);
}
fclose(in); //Closes the file//

/* The first function call sends all the lake temperature arrays to the function
 * highest_temp to determine the highest temperature of all the lakes. The same
 * The process is repeated for the function coldest_temp. Both of these functions also
 * return which lake had the highest temperature and the day it occurred on. Then these
 * dates are sent to the function 'date' which changes the day into the standard
 * day/month format.
 */
warmest_temp = highest_temp(sup_temp, mich_temp, huron_temp, erie_temp, ont_temp, stclr_temp,
365, &day_of_year, names_of_lakes, lake);
strcpy(lake_warm, lake);
warmest_day = date(day_of_year, month);
strcpy (month_high, month);
coldest_temp = lowest_temp(sup_temp, mich_temp, huron_temp, erie_temp, ont_temp, stclr_temp, 365,
&day_of_year, names_of_lakes, lake);
strcpy(lake_cold, lake);
coldest_day = date(day_of_year, month);
strcpy (month_low, month);

printf("%57s%d\n", "Warmest Overall Temperature of All the Lakes for ", year);
printf("-----\n");
printf("Lake %s: %s %d, %.2lf degrees Celsius\n\n\n", lake_warm, month_high, warmest_day,
warmest_temp);
printf("%57s%d\n", "Coldest Overall Temperature of All the Lakes for ", year);
printf("-----\n");
printf("Lake %s: %s %d, %.2lf degrees Celsius\n\n\n", lake_cold, month_low, coldest_day,
coldest_temp);
return 0;
}

```

Part 5 and Part 6 codes (merged together) :

```

/* CPS 125 Term Project - Parts 5 and 6 */
/* Name: Reza Aablue */
/* Section: 10 */
/* Student Number: 500966944 */

#include <stdio.h>
#include <stdlib.h>
#include <math.h>
int main () {
    /* The strategy/approach to this part of the term project is to store the

```


data of all 6 lakes into separate arrays for each lake and by using iteration (for loops), the summer and winter averages for the lakes will be calculated. Next, the lakes' summer and winter temperature averages will be sorted out in descending order. */

```
/* All variables and arrays will be declared here. */
FILE* infile;
int year, day;
double superior_array[365];
double michigan_array[365];
double huron_array[365];
double erie_array[365];
double ontario_array[365];
double saint_clair_array[365];
double temp1, temp2, temp3, temp4, temp5, temp6;
int counter;
int i;

int summer_counter, winter_counter_1, winter_counter_2;
int summer_days=0, winter_days=0;

double summer_sum_sup, summer_sum_mich, summer_sum_hur, summer_sum_erie,
summer_sum_ont, summer_sum_stc;
double summer_avg_sup, summer_avg_mich, summer_avg_hur, summer_avg_erie,
summer_avg_ont, summer_avg_stc;

double winter_sum_sup, winter_sum_mich, winter_sum_hur, winter_sum_erie, winter_sum_ont,
winter_sum_stc;
double winter_avg_sup, winter_avg_mich, winter_avg_hur, winter_avg_erie, winter_avg_ont,
winter_avg_stc;

double summer_averages[6], winter_averages[6];
double temporary_variable, temporary_variable_2;
int j, k, L; /* Used for the sorting algorithm. */

/* Opening the data file and using a while loop to collect
and store data for each lake into designated arrays. Then,
the file is closed when all data has been scanned and sorted into
their respective arrays. */
system("color f0");

infile=fopen("LakedataTermProject.txt", "r");
if (infile==NULL)
```

```

        {printf("\nThere was an error with opening the file.");}
else
    {while (!feof(infile))
        {fscanf(infile, "%d", &year);
        fscanf(infile, "%d", &day);

        fscanf(infile, "%lf", &temp1);
        superior_array[counter]=temp1;

        fscanf(infile, "%lf", &temp2);
        michigan_array[counter]=temp2;

        fscanf(infile, "%lf", &temp3);
        huron_array[counter]=temp3;

        fscanf(infile, "%lf", &temp4);
        erie_array[counter]=temp4;

        fscanf(infile, "%lf", &temp5);
        ontario_array[counter]=temp5;

        fscanf(infile, "%lf", &temp6);
        saint_clair_array[counter]=temp6;
        ++counter;}

fclose(infile);

/* Using a for loop to sum up all the summer temperatures for the 6 lakes (days 172-265). */
for (summer_counter=171; summer_counter<=264; ++summer_counter)
    {summer_sum_sup+=superior_array[summer_counter];
    summer_sum_mich+=michigan_array[summer_counter];
    summer_sum_hur+=huron_array[summer_counter];
    summer_sum_erie+=erie_array[summer_counter];
    summer_sum_ont+=ontario_array[summer_counter];
    summer_sum_stc+=saint_clair_array[summer_counter];
    ++summer_days;}

/* Using a for loop to sum up all the winter temperatures for the 6 lakes (days 1-79). */
for (winter_counter_1=0; winter_counter_1<=78; ++winter_counter_1)
    {winter_sum_sup+=superior_array[winter_counter_1];
    winter_sum_mich+=michigan_array[winter_counter_1];
    winter_sum_hur+=huron_array[winter_counter_1];
    winter_sum_erie+=erie_array[winter_counter_1];
    winter_sum_ont+=ontario_array[winter_counter_1];

```

```

        winter_sum_stc+=saint_clair_array[winter_counter_1];
        ++winter_days;}

/* Using a for loop to sum up all the winter temperatures for the 6 lakes (days 355-365). */
for (winter_counter_1=354; winter_counter_1<=364; ++winter_counter_1)
    {winter_sum_sup+=superior_array[winter_counter_1];
    winter_sum_mich+=michigan_array[winter_counter_1];
    winter_sum_hur+=huron_array[winter_counter_1];
    winter_sum_erie+=erie_array[winter_counter_1];
    winter_sum_ont+=ontario_array[winter_counter_1];
    winter_sum_stc+=saint_clair_array[winter_counter_1];
    ++winter_days;}

/* Calculating summer temperature averages for the 6 lakes. */
summer_avg_sup=summer_sum_sup/summer_days;
summer_averages[0]=summer_avg_sup;

summer_avg_mich=summer_sum_mich/summer_days;
summer_averages[1]=summer_avg_mich;

summer_avg_hur=summer_sum_hur/summer_days;
summer_averages[2]=summer_avg_hur;

summer_avg_erie=summer_sum_erie/summer_days;
summer_averages[3]=summer_avg_erie;

summer_avg_ont=summer_sum_ont/summer_days;
summer_averages[4]=summer_avg_ont;

summer_avg_stc=summer_sum_stc/summer_days;
summer_averages[5]=summer_avg_stc;

/* Calculating winter temperature averages for the 6 lakes. */
winter_avg_sup=winter_sum_sup/winter_days;
winter_averages[0]=winter_avg_sup;

winter_avg_mich=winter_sum_mich/winter_days;
winter_averages[1]=winter_avg_mich;

winter_avg_hur=winter_sum_hur/winter_days;
winter_averages[2]=winter_avg_hur;

winter_avg_erie=winter_sum_erie/winter_days;

```

```

winter_averages[3]=winter_avg_erie;

winter_avg_ont=winter_sum_ont/winter_days;
winter_averages[4]=winter_avg_ont;

winter_avg_stc=winter_sum_stc/winter_days;
winter_averages[5]=winter_avg_stc;

/* Sorting algorithm for the summer averages data set. */
for (j=0; j<4; ++j)
    {for (i=0; i<6; ++i)
        {if (summer_averages[i]<summer_averages[i+1])
            {temporary_variable=summer_averages[i+1];
             summer_averages[i+1]=summer_averages[i];
             summer_averages[i]=temporary_variable;}
        }
    }

/* Printing out the temperature averages of the two seasons in descending order. */
printf("The temperature averages are listed from warmest to coldest.\n");
printf("\nAverage Summer Temperatures amongst the 6 lakes (Degrees Celsius/day)");
printf("\n-----");
printf("\n1. %.2lf (Lake Erie)", summer_averages[0]);
printf("\n2. %.2lf (Lake St. Clair)", summer_averages[1]);
printf("\n3. %.2lf (Lake Ontario)", summer_averages[2]);
printf("\n4. %.2lf (Lake Michigan)", summer_averages[3]);
printf("\n5. %.2lf (Lake Huron)", summer_averages[4]);
printf("\n6. %.2lf (Lake Superior)", summer_averages[5]);

printf("\n\nAverage Winter Temperatures amongst the 6 lakes (Degrees Celsius/day)");
printf("\n-----");
printf("\n1. %.2lf (Lake Ontario) \n2. %.2lf (Lake Michigan) \n3. %.2lf (Lake Huron) \n4. %.2lf (Lake Superior) \n5. %.2lf (Lake Erie) \n6. %.2lf (Lake St. Clair)\n", winter_avg_ont, winter_avg_mich, winter_avg_hur, winter_avg_sup, winter_avg_erie, winter_avg_stc);

return(0);
}
}

```

Part 7 and Part 8 codes (merged together) :

```

/* CPS 125 Term Project - Parts 7 and 8*/
/* Name: Aram Ebadi Fard Azar */
/* Section: 11 */
/* Student Number: 500963606 */

```

```

/* Due Date: April 13, 2020 */
#include <stdio.h>
#include <stdlib.h>
int main ()
{
    system ("color f0");
    int above_twenty[6]={0,0,0,0,0,0}, below_zero[6]={0,0,0,0,0,0}; //Declaring and initializing the array
    FILE *f = fopen("glsea-temps2019_1024.dat.txt", "r"); //Opening 2019 data file
    if (f == NULL)
    {
        printf("no such file."); // if file was not in the right directory, then this message will be printed
        return 0;
    }

    int year[365]; // Initializing each array
    int days[365];
    double michigan_array[365];
    double superior_array[365];
    double huron_array[365];
    double erie_array[365];
    double ontario_array[365];
    double saint_clair_array[365];

    int i = 0;
    while (!feof(f))
    {
        fscanf(f,"%d %d %lf %lf %lf %lf %lf %lf", &year[i], &days[i],
        &superior_array[i],&michigan_array[i], &huron_array[i], &erie_array[i], &ontario_array[i],
        &saint_clair_array[i]);
        I++;} //Scanning for the elements inside the arrays

    //Incrementing the array elements if temp > 20 or < 0
    for (int i = 0; i < 365; i++) //For loop to go through columns
    {
        //Part 7
        if (superior_array[i] > 20) //compares the scanned array to 20
            above_twenty[0]++;
        //Part 8
        if (superior_array[i] < 0)
            below_zero[0]++;
        //Part 7
        if (michigan_array[i] > 20)
            above_twenty[1]++;

```

```

//Part 8
if (michigan_array[i] < 0)
    below_zero[1]++;
//Part 7
if (huron_array[i] > 20)
    above_twenty[2]++;
//Part 8
if (huron_array[i] < 0)
    below_zero[2]++;
//Part 7
if (erie_array[i] > 20)
    above_twenty[3]++;
//Part 8
if (erie_array[i] < 0)
    below_zero[3]++;
//Part 7
if (ontario_array[i] > 20)
    above_twenty[4]++;
//Part 8
if (ontario_array[i] < 0)
    below_zero[4]++;
//Part 7
if (saint_clair_array[i] > 20)
    above_twenty[5]++;
//Part 8
if (saint_clair_array[i] < 0)
    below_zero[5]++;
}

//Array for storing the names of each lake
const char *lakes[6] = {"Superior", "Michigan", "Huron", "Erie", "Ontario", "St.Clair"};

printf("The number of days in 2019 that one could swim comfortably in the great six lakes,
assuming that the temperature above 20 degrees was ideal for swimming \n\n");

//Printing a counter of the # of times each lake is > 20 & < 0
for (int i = 0; i < 6; i++)
{
    printf("Lake %s\nAbove 20 degrees: %d, Below 0 degrees: %d\n", *(lakes+i), above_twenty[i],
below_zero[i]);
}

return(0);

```

```
}
```

Part 9 code:

```
/*Term Project Part 9*/
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
int main(void)
```

```
{
```

```
    system ("color f0");
```

```
    int i,j;
```

```
    double temp[365][8]; //initializing the array
```

```
    double
```

```
sum_sup=0,avg_sup=0,sum_mich=0,avg_mich=0,sum_huron=0,avg_huron=0,sum_erie=0,avg_erie=0,sum_ont=0,avg_ont=0,sum_st_clr=0,avg_st_clr=0,total_avg;
```

```
    double
```

```
temp2[365][8],sum_sup2=0,avg_sup2=0,sum_mich2=0,avg_mich2=0,sum_huron2=0,avg_huron2=0,sum_erie2=0,avg_erie2=0,sum_ont2=0,avg_ont2=0,sum_st_clr2=0,avg_st_clr2=0,total_avg2;
```

```
    FILE*in=fopen("data_2019.txt", "r"); //opening the 2019 data file
```

```
    FILE*in2=fopen("data_2018.txt", "r"); //opening the 2018 data file
```

```
    for (i=0;i<365;i++) //For loop to go through rows
```

```
    {
```

```
        for (j=0;j<8;j++) //For loop to go through columns
```

```
        {
```

```
            fscanf(in,"%lf",&temp[i][j]); //scanning elements in rows and columns for each
```

```
file
```

```
            fscanf(in2,"%lf",&temp2[i][j]);
```

```
        }
```

```
        sum_sup=sum_sup+temp[i][2]; //Finding the averages for each column in 2019 file, then
```

```
2018 file
```

```
        avg_sup=sum_sup/365;
```

```
        sum_sup2=sum_sup2+temp2[i][2];
```

```
        avg_sup2=sum_sup2/365;
```

```
        sum_mich=sum_mich+temp[i][3];
```

```
        avg_mich=sum_mich/365;
```

```
        sum_mich2=sum_mich2+temp2[i][3];
```

```

    avg_mich2=sum_mich2/365;

    sum_huron=sum_huron+temp[i][4];
    avg_huron=sum_huron/365;

    sum_huron2=sum_huron2+temp2[i][4];
    avg_huron2=sum_huron2/365;

    sum_erie=sum_erie+temp[i][5];
    avg_erie=sum_erie/365;

    sum_erie2=sum_erie2+temp2[i][5];
    avg_erie2=sum_erie2/365;

    sum_ont=sum_ont+temp[i][6];
    avg_ont=sum_ont/365;

    sum_ont2=sum_ont2+temp2[i][6];
    avg_ont2=sum_ont2/365;

    sum_st_clr=sum_st_clr+temp[i][7];
    avg_st_clr=sum_st_clr/365;

    sum_st_clr2=sum_st_clr2+temp2[i][7];
    avg_st_clr2=sum_st_clr2/365;
}

printf(" \n-> Below is the yearly average temperature (degrees C) for each of the lakes:\n\n");
printf(" ~~~~~~\n");
printf(" | 2019 average Temp. | 2018 average Temp. |\n");
printf(" |-----| \n");
printf(" |Lake Superior : %.2lf |",avg_sup);
printf(" Lake Superior : %.2lf\n",avg_sup2);

printf(" |Lake Michigan : %.2lf |",avg_mich);
printf(" Lake Michigan : %.2lf\n",avg_mich2);

printf(" |Lake Huron   : %.2lf |",avg_huron);
printf(" Lake Huron   : %.2lf\n",avg_huron2);

printf(" |Lake Erie     : %.2lf |",avg_erie);
printf(" Lake Erie     : %.2lf\n",avg_erie2);

```



```

printf(" |Lake Ontario : %.2lf |",avg_ont);
printf(" Lake Ontario : %.2lf\n",avg_ont2);

printf(" |Lake St.Clair : %.2lf |",avg_st_clr);
printf(" Lake St.Clair : %.2lf\n",avg_st_clr2);

total_avg = (avg_sup + avg_mich + avg_huron + avg_erie + avg_ont + avg_st_clr)/6;
total_avg2 = (avg_sup2 + avg_mich2 + avg_huron2 + avg_erie2 + avg_ont2 + avg_st_clr2)/6;

printf(" ~~~~~~\n\n");

printf("-> 2019 average temperature for all the six lakes: %.2lf (degrees C).\n\n",total_avg);
printf("-> 2019 average temperature for all the six lakes: %.2lf (degrees C).\n\n",total_avg2);

fclose(in);
fclose(in2);

return(0);

}

```