Homework #9

Processing Temperature Readings from a File

Due: November 27 by 11:59:59 PM **Assigned:** November 15, 2018

Write a C++ program which reads data from a file and processes it as listed in the Requirements. Download the file "temperature.dat" from Blackboard. Do NOT change the name of the file. This file contains a week of hourly temperature data with one line for each of the 7 days, where each line contains 24 integers (separated by spaces) which are the temperatures in Fahrenheit for each hour of the day. This is a text file and you can open it and view the data

Requirements:

- Name your source file `program9.cpp`
- Make an array with the definition:

  ```
  double celcius[7][24];
  ```

  Read in the data from the file `temperature.dat`, convert each value to celcius, and save the celcius temperatures in this array.
- Write a function with the following prototype:

  ```
  void show_warmer(double temperature[][24], int days, double cutoff);
  ```

  This function takes a parameter `temperature` which is a two-dimensional array of some number of days, of 24 hours each. The second parameter `days` is the number of days for which to access the array `temperature`. The third parameter `cutoff` is a value which elements of `temperature` are compared to in looking for values which are larger. This function needs to print out the values which it finds in the format of the example below (e.g. the output line "At day 2, hour 19, the temperature was 22.78 C.").
- Prompt the user to enter a value, and then call the function `show_warmer()` using the value entered by the user for the `cutoff` parameter. (The prompts and layout must look like the example below.)
- Write a function with the following prototype:

  ```
  double find_average(double temperature[][24], int days);
  ```

  This function takes a parameter `temperature` which is a two-dimensional array of some number of days, of 24 hours each. The second parameter `days` is the number of days for which to access the array `temperature`. This function needs to find and return the average of all the values in `temperature`. It must **not** print anything on the screen.

- Call the function `find_average()` (after calling `show_warmer()`). The main function must print the result obtained from `find_average()` in the format of the example below (e.g. the output line "The average temperature over all the days was: 18.66 C.").
- Write a function with the following prototype (*Multiple lines not required, this was just to fit it on the page*):

```
void find_daily_mmm(double day_temps[],
                    double &max,
                    double &min,
                    double &mean);
```

  The first parameter `day_temps` is an array of 24 values (temperatures for one day) which is the input to the function. The remaining three parameters are reference parameters because they will be used to store output values calculated by the function. After running the function, `max()` needs to contain the maximum value of `day_temps`, `min` the minimum value, and `mean` the mean value.

- Write a loop which calls the function `find_daily_mmm()` for each day of `celcius` and then prints the values it found in the format shown below (e.g. prints lines like "Day 1: max 21.11 C, min 13.33 C, mean 17.75 C.").
- Temperature values displayed in the output must be shown with 2 decimal places.

Continued on next page

- A sample run of your program should look like:

```
Enter the value for which to find warmer temperatures (C): 22.3

Times at which temperatures warmer than 22.30 C were found:

At day 2, hour 19, the temperature was 22.78 C.
At day 4, hour 17, the temperature was 22.78 C.
At day 4, hour 18, the temperature was 23.33 C.
At day 4, hour 19, the temperature was 22.78 C.
At day 7, hour 19, the temperature was 22.78 C.

The average temperature over all the days was: 18.66 C.

The maximum, minimum, and mean temperatures for each day:

Day 1:  max 21.11 C,  min 13.33 C,  mean 17.75 C.
Day 2:  max 22.78 C,  min 15.56 C,  mean 18.40 C.
Day 3:  max 22.22 C,  min 16.67 C,  mean 19.05 C.
Day 4:  max 23.33 C,  min 17.78 C,  mean 19.79 C.
Day 5:  max 21.11 C,  min 16.11 C,  mean 18.61 C.
Day 6:  max 19.44 C,  min 16.11 C,  mean 18.15 C.
Day 7:  max 22.78 C,  min 15.56 C,  mean 18.87 C.
```

Hints:

- Remember to check that the file was successfully opened, and to close it when you're done with it
- Remember to set the display format for floating point numbers to 2 decimal places, as shown in the example above.
- Note that the day/hour numbers shown in the output start at 1, so are 1 higher than the array index which the data comes from.

Reminders:

- Be sure that your program includes your name, ID, description, etc. as shown in the General Homework Requirements Handout
- Use good style including indentation, comments, etc. Part of the grade will be for style and quality.
- Carefully test your program.
- You are welcome to write your program at home. If you do, be sure to compile and test it in the lab before submitting it.

How to submit your program:

- Submit the file `program9.cpp` electronically using the following terminal command:
  For the 12:30 lecture section:
  `~cs211a/bin/handin 9 program9.cpp`

  For the 5:35 lecture section:
  `~cs211b/bin/handin 9 program9.cpp`