jikj Program 6

CS2010 25 pts

Spring, 2018 Due: 11:59pm, Wednesday, April 4, 2018

**Problem:** Is the Earth warming up? Are the temperatures we have been experiencing recently normal fluctuations or evidence of a warming trend? Write a C++ program to analyze temperature data from March, April and May of 2015 for Ohio. Create a project using your last name, first initial and pgm6 for the project folder and .cpp file names (e.g., **WaveH\_pgm6** and **WaveH\_pgm6.cpp**). Your output may be displayed on the screen.

**Input:** The temperature (in Celsius) for each day of March, April and May (92 temperatures) is stored in the file pgm6.txt. Copy this file from the Canvas class web site to your project folder. Your project should read in each temperature and store it in the next available element of an array of type *double*. All of the temperatures must be stored in the same array. Then using the data values stored in the array, perform the steps described below.

**Processing:** Your *main* function should do the following:

1. Declare the array to hold the temperatures and any other variables needed.

2. Call a function to read the daily temperatures from the file **pgm6.txt**,storing each temperature in an element in the array. Open and close the data file in this function. All further processing must be done using the data values stored in the array. Temperatures in the file look like this:

0

-5.55556

-1.66667

6.11111

3. Call a function to display the temperatures for the month of March. Include a heading with your name and two columns – one showing the day of the month and one showing the temperature for that day (see sample below).

Temperatures for March

Your Name

Day Temperature

of month (Celsius)

-----------------------

1 0

2 -6

3 -2

4 6

4. Call a function to convert all of the temperatures in the array from Celsius to Fahrenheit (F = 9.0/5.0 \* C + 32), replacing the original Celsius temperature with the Fahrenheit value for each element. Note: You do not need to create a new array for this function, just change the values in the original array.

5. Call a function to display the temperatures for the month of April to show that the temperatures have been converted correctly. Include a heading, your name and the two columns – see sample below. (See if you can use the function from Step 3 again.)

Temperatures for April

Your Name

Day Temperature

of month (Fahrenheit)

----------------------------

1 61

2 73

3 57

4 56

6. Call a function to find and return the average temperature in Fahrenheit for the three-month period. The *main* function should display this result with an appropriate description.

7. Call a function to find and display the number of days that had a temperature above the average for this period. Clearly label your output.

8. Call a function to find the highest temperature for each month and display each value with an appropriate label. Hint: Start by finding the highest temperature for March. Then think about how you could generalize this to do the same for the other two months.

**Program Design:** A structure chart for your program is shown below. Work on one function at a time. For each function decide what its parameter(s) should be: what data, if any, needs to be passed to the function; what results, if any, will the function return or pass back. Then write the function prototype, function call and function definition. Make sure these match in the required ways. Determine where each of these should be located in your program and type them in, including documentation. Be sure to test and debug each function before going on to the next one.

*main*

readTemps

findHiTemp

displayTemps

findDaysAboveAvg

convertToFahr

findAvgTemp

**Program Documentation & Style:**

1. Declare all constants and variables that your program uses at the beginning of your program.

2. Your program should include three types of comments. BE SPECIFIC!

a. Header comments at the beginning of your program including lines with:

- Your name, course name, and class time

- Program assignment number, program file name and due date

- **Purpose**: a sentence or two explaining the purpose of the program

- **Input**: a description of the input data needed by the program when you run it

- **Processing**: a description of the processing (calculations) done by the program

- **Output**: a description of the results (output) produced by the program

b. In-line comments: There should be an in-line comment for each main step in your program. In general, this means at least one comment with each group of C++ statements that handles the input, the processing and the output steps of your program.

c. Function comments: Include a sentence or two before each function definition explaining the purpose of the function.

3. Use meaningful identifier names

4. Include clear prompts for the user about entering the data.

5. Include clear descriptions of the results when you display them.

**CS2010 Program 6 Grading Rubric**

**Due**: 11:59pm, Wednesday, April 4, 2018

\_\_\_\_\_\_\_ Project named correctly and .cpp file turned in on Canvas.

\_\_\_\_\_\_\_ (3) Header comments complete, including labeled sections identifying the Purpose, Input, Processing and Output

\_\_\_\_\_\_\_ (3) Function, in-line comments complete, uses good program style (meaningful data-names, whitespace, indentation)

\_\_\_\_\_\_\_ (3) Function to read in temperatures from file correct. Uses parameter passing correctly. Function prototype included. Input file opened/closed in this function.

\_\_\_\_\_\_\_ (4) Function to display temperatures for month correct. Uses parameter passing correctly. Function prototype included.

\_\_\_\_\_\_\_ (3) Function to convert temperatures from Celsius to Fahrenheit correct. Uses parameter passing correctly. Temperatures for April displayed after conversion. Function prototype included.

\_\_\_\_\_\_\_ (3) Function to find average temperature correct. Uses parameter passing correctly. Function prototype included.

\_\_\_\_\_\_\_ (3) Function to count number of days above average temperature correct. Uses parameter passing correctly. Function prototype included.

\_\_\_\_\_\_\_ (3) Function to find the highest temperature for each month correct. Uses parameter passing correctly. Function prototype included.

\_\_\_\_\_ **Bonus** 2 pts. Use recursion to display the array temperatures.

\_\_\_\_\_\_\_ **(25) Total Points (27 with Bonus)**