HW: Temperature Queries (Part 2)

Points

Points	
85	Working Program
15	Code Review
100	**TOTAL**

Submission

MAXIMUM NUMBER OF SUBMISSIONS: 18
*If you need more submissions than that, we **will** give you additional submissions during office hours without penalty, provided that you can illustrate that you've tested your program significantly locally (this notice posted on Sun, 25 Nov 2018 15:13:51-0500; additional submissions were granted to those who submitted something to Mimir prior to this). If you haven't tested significantly locally, we will request that you do before additional submissions are awarded.* ***Remember, you can see the test cases in Mimir; this means you can execute them locally too.***

1. **Source Code**: Submit the source code (**LinkedList.h, LinkedList.cpp, Node.h, TemperatureData.h, TemperatureData.cpp, TemperatureDatabase.h, TemperatureDatabase.cpp, and main.cpp** files) to Mimir

Specifications

- Part 1: You will implement a linked list to organize temperature sensor readings from a file.
- Part 2: Process a file with queries to report based on the data.

Design (Optional)

1. Algorithm for determining the average as specified.
2. Algorithm for determining the mode as specified.
3. Additional Components: Test cases

Program

You will implement a program that receives as input two files:

1. **A temperature file (e.g., temps.dat)**

   See Part 1.

2. **A query file (e.g., queries.dat)**

   This file contains a list of queries (i.e., requests for information) that your program should calculate based on the data in file temps.dat. Each line in the file contains one query. There are two types of queries:

   - **AVG:** given a location (specified as a station id) and a range of years (specified as two integers, for example, 2000 2006), it computes the average temperature for the location and period. If no data is available for the location/period, the answer will be the string “unknown”.
   - ***MODE:*** *given a location and a range of years, it identifies the most common rounded temperature value in the period.*
     i. If no data is available for the period, the expected result is the string “unknown”.
     ii. If more than one mode, you should return the one with the highest value.
     iii. Notice that the temperature values in file temps.dat are double numbers. You should round these values to the nearest integer before computing the mode. For example, 8.03, 8.1, and 8.5 are all rounded to 8; 8.51 and 8.8 are rounded to 9. So with this set of

data, there are three eights and two nines. And the mode would be 8.

Your program should read these two files and generate a file result.dat with the queries and their results. Regardless,

- temps.dat might look like this:

```
411048 2015 1 9.58
411048 2015 3 14.82
411048 2016 4 20.51
411048 2016 1 10.99
411000 1973 1 0.54
411048 2016 3 18.40
411048 2016 5 -99.99
```

- queries.dat might look like this:

```
411048 AVG 2015 2016
411138 AVG 1995 1995
411048 MODE 2015 2016
```

Then your program is expected to produce a file named “result.dat” with the following content:

```
411048 2015 2016 AVG 14.86
411138 1995 1995 AVG unknown
411048 2015 2016 MODE 21
```

Requirements

- Start with the code you completed for Part 1.

- Your program should receive the names of the input files as command line arguments. The first argument will be the temperature file, the second the queries file.
- If the input files contain a line with an invalid format, you should output on the console an error message and finish executing the program.
    - Do not throw an exception.
    - Output the message in the console beginning “Error: “ followed by the description shown below followed by the invalid value. For example:
        - Data File
            - Error: Invalid temperature -1221.11
            - Error: Invalid year 2020
            - Error: Invalid month 0
            - Error: Unable to open input.dat
            - Error: Other invalid input
        - Query File
            - Error: Invalid range 1996-1995
            - Error: Unsupported query MAX
            - Error: Other invalid query
            - ~~*Note: You can have an invalid year in the query so that error can occur here too.*~~
            - *Invalid years will be part of an invalid range.*
                - Error: Invalid range 1776-2025

- Examples of lines with an invalid format for queries.dat:

```
AVG 2015 2016
411138 AVG 1996 1995
411048 MODE 1500 2016
411048 MAX 2015 2016
```

- The output file created by your program should be named **result.dat**.

Grading

1. We will unit test TemperatureDatabase performQuery() function. We will look for error messages.
2. We will test the entire program, look for error messages, and examine your result.dat file.

Sample Input/Output Files

Note: these are only sample input/output files; we expect you to construct additional test sets.

- Get input files to build list
- Get Query files
- Get Result files

Hints & Comments

- It is often a good idea to develop your solution incrementally, completing and testing components of your overall program as you develop them.
- When you round a double value to calculate the mode, make sure that you are not just truncating the value: the rounded value for 10.6 should be 11, not 10.
- Notice that, as specified, the program you wrote is not very useful, as we often are not interested in average over years. We would prefer to be able to compare averages of specific months across year ranges. Once you complete your program, think about what you would have to do to be able to provide averages for specific months.
- If you tried your solution with very large input files, you may have noticed that your solution takes some time to provide the answers. Students in the Computer Science and Computer Engineering majors have the opportunity to take many other courses (data structures, databases, computer systems, operating systems, data mining, distributed systems, parallel computing) where they learn concepts and techniques to achieve efficient queries over large datasets.