Project 3
CS221: C and Systems Programming – Fall 2018

Deadline: November 25, 2018 at 11:59pm

Atmospheric Data Analysis

This project gives you the opportunity to put your C skills to use. We will be analyzing data from the National Oceanic and Atmospheric Administration (NOAA) North American Mesoscale Forecast System to learn more about the climate in a few different states.

C is a great match for data analysis, at least in the speed department: when youre processing millions of lines of data, youll be able to get things done much faster.

```
./climate data_tn.tdv data_wa.tdv

Opening file: data_tn.tdv
Opening file: data_wa.tdv
States found: TN WA
-- State: TN --
Number of Records: 17097
Average Humidity: 49.4%
Average Temperature: 58.3F
Max Temperature: 110.4F on Mon Aug  3 11:00:00 2015
Min Temperature: -11.1F on Fri Feb 20 04:00:00 2015
Lightning Strikes: 781
Records with Snow Cover: 107
Average Cloud Cover: 53.0%
-- State: WA --
Number of Records: 48357
Average Humidity: 61.3%
Average Temperature: 52.9F
Max Temperature: 125.7F on Sun Jun 28 17:00:00 2015
Min Temperature: -18.7F on Wed Dec 30 04:00:00 2015
Lightning Strikes: 1190
Records with Snow Cover: 1383
Average Cloud Cover: 54.5%
```

Testing Your Code

There are three data files included to test your code:

- `data_tn.tdv`
- `data_wa.tdv`
- `data_multi.tdv.gz`

`data_multi` is compressed to save space. To decompress it, use your favorite archive utility or the command line:

```
gunzip data_multi.gz
```

Each file contains one record per line with fields separated by tab characters ('\t'). The columns are organized as follows:

```
TN    1424325600000    dn20t1kz0xrz    67.0    0.0    0.0    0.0    101872.0    262.5665
TN    1422770400000    dn2dcstxsf5b    23.0    0.0    100.0  0.0    100576.0    277.8087
TN    1422792000000    dn2sdp6pbb5b    96.0    0.0    100.0  0.0    100117.0    278.49207
TN    1422748800000    dn2fjteh8e80    6.0     0.0    100.0  0.0    100661.0    278.28485
TN    1423396800000    dn2k0y7ffcup    14.0    0.0    100.0  0.0    100176.0    282.02142
```

...

We will also test your programs with other input files. Note that you can assume that each line in the files will contain all the fields. No need to check for malformed files or lines.

Fields:

1. State code (e.g., CA, TX, etc)
2. Timestamp (time of observation as a UNIX timestamp)
3. Geolocation (geohash string)
4. Humidity (0 - 100
5. Snow (1 = snow present, 0 = no snow)
6. Cloud cover (0 - 100
7. Lightning strikes (1 = lightning strike, 0 = no lightning)
8. Pressure (Pa)
9. Surface temperature (Kelvin)

Hints and Resources

The dataset contains temperatures in Kelvin rather than degrees Fahrenheit. To convert K to F, you can use the following formula:

```
deg_f = deg_k * 1.8 - 459.67;
```

The times the measurements were taken are expressed as `Unix` timestamps. These can be converted to string form with the `ctime` function. You will also need to divide the timestamps in the data files by 1000 to adjust for the precision `ctime` expects:

```
#include <time.h>

timestamp = timestamp / 1000;
printf("Time: %s", ctime(&timestamp));
```

Finally, be careful when determining which C data types to use in your struct. If you are wondering what can be stored in different data types, check Wikipedias page on C Data Types.

Grading

Your code must compile and run correctly on the Linux lab machines. If we cannot compile your code on the lab machines, you will receive no credit.

Feature	Points
Correct climate statistics	30
Error handling (missing files, etc).[1]	15
Support for processing multiple files	15
Function documentation and comments	15
Correct formatting and unit conversions	10
Program Usage Message	10
Coding Style	5

[1] You can assume the data files we provide do not have any malformed data or missing fields.

Submission Guidelines:
Prior to the deadline, upload one zip file containing only two files: your climate.c source code, and status.txt, to Canvas. The ZIP file name must be in the following format: LastName_FirstName_StudentID_proj3.zip For instance if my student ID is 123456789 and I am submitting my solution for project 3, then I am going to compress status.txt and climate.c, and rename the zip file to: Pournaghshband_Vahab_123456789_proj3.zip Do not submit/include any other file such as the executable file. A good sanity check is to check your zip file for corruption by extracting (unzipping) it and testing whether it did compress it successfully. If we cannot unzip your submission, you will receive no credit.

Sample status.txt file:

```
Vahab Pournaghshband - Project 3 The program works as required. It compiles/runs and the
output matches the correct format to the letter. However, the style and formatting is incorrect
because I DIDNT comment it (didnt even put my name in the file).
```