Computer Science 327
Homework 3

This assignment builds on Assignment 2 by providing a query interface to the user for analyzing a data set read by your weather data I/O functions. The format for these queries is defined below. Your program will input these queries from the user via the console keyboard (stdin) or via a filename supplied by the command line parameters. Output is displayed on the standard output device, or to a file that may be specified on the command line. The command line format for input and output files is also defined below.

Date and time are specified by the following format:

mm/dd/yyyy-hh:mm

where "mm" is the two-digit month starting with January as 01, "dd" is the two-digit day, "yyyy" is the four-digit year, "hh" is the two-digit hour in 24-hour format (0, 23), and "mm" is the minutes past the hour. Example of valid data time strings are:

05/21/2014-23:14

Note that 4/4/2014-9:33 is not a valid format.

There are four different kinds of queries:

```
histogram <header> <bins> <start date/time> <end date/time>
```

Create a table of values that represents a histogram of the data. Bins is an integer that represents the number of equally spaced bins to use in the histogram between the minimum and the maximum elements of data in the data set for the column named in header. Example:

```
histogram Temp 5 01/01/2014-00:00 12/31/2014-23:59
```

The above would output a table (example shown below) of a histogram of the temperature column with 5 bins equally spaced between the minimum and maximum temperatures between the start date and time and the end date and time, inclusive.

Example table output:

```
Temp range          Number of occurrences
-10.2 : -1.2        210
-1.2 : 7.8          343
7.8 : 17.8          542
17.8 : 26.8         903
26.8 : 35.8         432
```

Note that a temperature of -1.2 is counted in the second bin and not the first. The last bin does take counts for both 26.8 and 35.8. Histogram data output is only valid for numerical only fields and should output an error message if otherwise.

```
min <header> <start date/time> <end date/time>
```

Computes the minimum value given by the column named by header between the start date/time and the end date/time. This query is only valid for numerical-only data columns. The output is displayed as follows given by an example:

```
Minimum = 32.4563
```

```
max <header> <start date/time> <end date/time>
```

Similar to min but displays maximum over the date/time range.

```
average <header> <start date/time> <end date/time>
```

Similar to min but displays the average over the data/time range.

Typing "quit" or receiving an EOF condition terminates the program.

The program operates with a command-line interface and optional command-line parameters as follows.

The program may take up to 3 command-line parameters. If more than three are present then the program will immediately terminate and give an error message, even if the first three parameters are valid. The command line parameters may come in any order and are distinguished as follows.

"df=<datafile>"
"if=<Query input file>"
"of=<Output file>"

If the "df" command line parameter is not given, the program will ask the user to enter a filename. If the "if" command line parameter is not given, then the program will read queries from stdin. If the "of" command line parameter is not given, then the program will output to standard output. If the file cannot be opened, the program will output an error message and exit. If a query contains a format error, the program will output an appropriate error message along with the query that failed, but will continue to process queries until a termination condition is found. Examples of command lines to start the program are:

```
analyze df=AmesWeatherData.csv if=q.txt of=output.txt
```

```
analyze df=AmesWeatherData.csv if=q.txt
```

```
analyze
```

```
analyze of=out.txt
```

For this assignment you are free to create any functions you wish and place them in any files. Part of your grade will be based on reasonable division of functionality between functions you use in your program. Proper division of functions in files will also be evaluated. You must create a valid makefile for your program, and the resulting executable file must be named "analyze".