CS 420
PROGRAM 7P

Purpose

The purpose of this assignment is to have you work with a little bit of everything that you've learnt so far.

Scenario

Earthquakes occur all over the world all the time. Though most of the earthquakes are never felt by us, they do get recorded in seismic laboratories all over the world. Seismic data analysts are interested to gather various statistics about these earthquakes and interpret them in some meaningful way to predict future earthquakes. Let us say there's an organization that wants to gather and analyze seismic data for certain regions in the U.S and Canada for a 31-day period in October 2014. Your job as a programmer would be to take in all the data that's given to you and come up with some specific statistics.

Method

To implement this program you will use several appropriate data structures based on what's needed. There is seismic data from several different regions coming from input files one for each region, they are all the same format and need to be stored in linked lists. A structure needs to be created to hold all the fields for each record in the dataset which becomes the data for each linked list entry. Since we want to keep track of all the regions that reported data, each linked list needs to be linked up in some manner.

Data

A sample record contains the following data as indicated below:

Year	Month	Day	Time (UTC)	Latitude	Longitude	Magnitude	Depth	Location
2014	10	28	06:10:35	37.041	97.930	5.0	21.3	4-mi-SW-of-Bluff-City,KS

Data Structures

Pay attention to the data type for each item and use appropriate declarations for each data item within the structure. *To store all the data from one data file you will need one linked list of structures – so if we had just one data file this would have been sufficient*. ***But we have multiple data files giving us data for multiple regions, therefore you need to have multiple linked lists to hold the entire data set.*** We will also need to connect up all the linked lists together by creating a linked list of "headers" for each list. A single global header will be sufficient to figure out where to reach out for the "linked list headers".

Requirements

1. The program should initially read all the data and store them in the appropriate structures
2. A summary of the data in each data file should be printed programmatically to an output file which includes the following pieces of information:
    a) Summary heading indicates range of Latitude and Longitude for the data set
    b) Total number of earthquakes recorded for the period
    c) Range of depth that earthquakes were recorded
    d) Range of magnitude of earthquakes
    e) Highest recorded magnitude with details of date, time and location

    The output file name is passed as the second command line argument, and is created and written to programmatically (not redirection of standard output using ">"). The output data file is needed only for this item (2).

3. The program should display a table that indicates day by day earthquake totals for all the datasets together. In other words a dynamically-allocated 2D array must be created of size Rx31 (or Rx32 if that's easier where R indicates the number of rows, one per region) whose cells contain the total number of earthquakes per day per region.
4. The program should display a table that indicates totals for magnitude ranges for each region. In other words a 2D array must be created of size Rx10 (if the highest magnitude for any region is 9). You can truncate magnitudes to the nearest integer just for the purposes of this table.
5. The program should display a table that indicates totals for depth ranges for each region. In other words a 2D array must be created of size RxA (look at sample output file for ranges to set your value for A for number of columns). You can truncate depths to the nearest integer just for the purposes of this table.
6. All dynamic memory allocated must be freed at the end of the program.

   All tables that are displayed for items (3), (4), (5) above must have clear row and column headings for ease of data analysis.

You are given the header file which contains the data structures and declarations of the functions needed (equake.h).You are also given the main module (equake.c). Make sure you understand how your functions need to be called. You will be submitting only the module containing your functions in equakeFunctions.c which needs to work with the main module supplied to you. You will also submit a makefile . You cannot make any changes to the header file or the main module.

Input

The input data is coming from several data files. There is one data file called data.txt supplied via command line arguments (via argc, argv) that actually contains all the other file names. All input values will come from the individual data files that are listed in data.txt. Here's how you would run the program: ./equakes data.txt summary.txt

The program will open and read the filenames from data.txt (one filename per line in data.txt) and in turn open up each file to process data. Here's a sample data.txt file containing names of other files:

```
Alaska.txt
Central.txt
InterMountain.txt
NorthEast.txt
NorthernCanada.txt
NorthWest.txt
```

Here are some records from one of the input files from a certain region:

```
2014 10 18 18:59:25 40.044 118.837 0.6 13.1 13km(8mi)-N-of-Trinity,NV
2014 10 18 18:06:23 39.461 110.945 1.3 6.0 14km(8mi)-N-of-Elmo,UT
2014 10 18 17:52:18 38.965 111.389 1.0 6.4 13km(8mi)-WN-of-Emery,UT
2014 10 18 17:16:18 41.861 119.617 1.8 0.8 11km(7mi)-NE-of-Wimer-Place,NV
2014 10 18 14:47:08 41.906 119.615 2.6 0.2 15km(9mi)-NE-of-Wimer-Place,NV
2014 10 18 14:27:03 41.874 119.618 2.0 0.0 12km(8mi)-NE-of-Wimer-Place,NV
2014 10 18 11:41:05 39.685 110.734 1.8 3.4 11km(7mi)-E-of-Helper,UT
2014 10 18 11:39:00 41.024 111.645 1.2 7.0 3km(2mi)-SE-of-Morgan,UT
```

Output

The output will first summarize the data as indicated above in item 2. The rest of the output is displayed on standard output. For items 4 and 5, my values are being truncated to integers so if you want your values to match with mine discard the fractional values (for example magnitude 4.1 and 4.7 will both be truncated to 4.0).

Sample output for displaying statistics for items 3, 4 and 5 are shown below:

**********EARTHQUAKE TOTALS FOR THE MONTH BY REGION**********

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
NorthEast.txt	2	3	3	4	1	1	0	1	3	4	1	1	1	3	0	0	3	1	1	0	1	2	2	2	0	3	2	1	1	2	1
Central.txt	6	15	9	7	9	8	16	9	11	8	6	11	13	4	5	11	18	8	5	12	9	11	4	8	6	5	2	5	5	8	3
InterMountain.txt	5	2	1	6	2	0	3	4	4	2	1	2	0	0	4	14	38	19	22	22	15	2	3	4	1	1	4	2	3	4	3
NorthWest.txt	15	18	12	14	9	12	14	15	5	8	8	7	10	4	7	10	9	0	5	9	9	5	4	7	1	6	6	6	4	8	4
NorthernCanada.txt	1	2	3	3	3	3	1	1	0	20	1	0	3	0	1	0	5	2	2	3	0	2	4	0	2	0	2	2	4	0	0
Alaska.txt	26	31	23	27	24	23	21	19	29	16	20	25	31	17	18	23	20	25	27	31	15	27	22	25	35	34	20	25	28	40	15

**********EARTHQUAKE TOTALS BY MAGNITUDE BY REGION**********

|Magnitude Range: 0 to 10 (increments of 1)

	0	1	2	3	4	5	6	7	8	9
NorthEast.txt	1	10	36	3	0	0	0	0	0	0
Central.txt	3	18	170	63	3	0	0	0	0	0
InterMountain.txt	33	100	51	7	2	0	0	0	0	0
NorthWest.txt	143	75	31	2	0	0	0	0	0	0
NorthernCanada.txt	0	6	30	26	7	1	0	0	0	0
Alaska.txt	0	610	136	13	3	0	0	0	0	0

**********EARTHQUAKE TOTALS BY DEPTH BY REGION**********

|Depth Range in KM (increments of 10)

	0	10	20	30	40	50	60	70	80	90	100+
NorthEast.txt	38	11	0	1	0	0	0	0	0	0	0
Central.txt	249	7	1	0	0	0	0	0	0	0	0
InterMountain.txt	163	30	0	0	0	0	0	0	0	0	0
NorthWest.txt	147	66	31	6	0	1	0	0	0	0	0
NorthernCanada.txt	31	36	1	2	0	0	0	0	0	0	0
Alaska.txt	186	118	48	41	50	25	52	36	49	41	116

Files Provided: equake.c, equake.h
Files to be submitted: equakeFunctions.c, makefile
(your makefile needs a target called equakeFunctions.o that builds the object file for equakeFunctions.c)

Data Structure representation: