CS 135
Exercise #12
Point value: 50
Date due: email source file (using mail utility) to your lab instructor by 11:59pm Tues, Nov 21

New Skills Practiced (Learning Goals)

- Problem solving and debugging.
- Filestreams.
- Arrays.

Design a program that will (in the main function)

- declare 2 arrays that can each store a maximum of 20 double type values
- declare the variables necessary to store a file name (string) and represent a file (ifstream)
- display your name, lecture and lab section #s,. and exercise # to the screen
- interactively prompt the user for the name of an input file and read the file name
- open the file (using an ifstream type variable to represent the file)
- read the content of the file
  - first value in the file will be an integer, n, that is greater than 0 and less than or equal to 20
  - n indicates how many values are to be stored in one of the arrays
  - read the n values and store them into one of the arrays
- close the input file
- call a void function that will be passed the 2 arrays (the array with the input values and the array that was not initialized) and the number of elements that were stored (n)
  - using the values in the input array, the function should do the following
    - double the value of each of the elements in the input array that have even subscripts (0,2,4,...) and store them in the corresponding locations of the second array
    - determine the cube root of each of the values in the input array that have odd subscripts (1,3,5,...) and store those values in the corresponding locations of the second array
    - print (to the screen) the content of both arrays (in 2 right justified columns with headings)
    - return to main
- call a value-returning function to sum all the values in the input array and then call the function again to sum the first int(n/2) values in the second array
  - the function should be designed to receive 2 parameters, an array of doubles and an integer, size
  - the function will add the first size values in the array and return the result
- print the sum of all the values in the input array with a label
- print the sum of the first int(n/2) values in the second array with a label

REQUIREMENTS

- Program must read input from a file using a filestream (no redirection).
- The input file can only be read 1 time.
- All header files referenced must be included.
- No global variables may be used.
- The void function must set the values for the 2nd array as described above and print the content of the 2 arrays in right justified columns (see example below) with 5 digits to the right of the decimal.
- Only one value returning function to sum the values can be implemented. It must be called twice.

ASSUMPTIONS

- The input file will contain a maximum of 20 double type values. The values may be positive, negative, or zero.
- Each value will be separated by blanks or linefeeds.
- The maximum width of a value to be displayed (including a negative sign) will be 11 columns.

NOTES:

- If you use library functions, make sure you include the appropriate header files.
- The cube root of a negative number is negative.
- It is a good idea to send a carbon copy to yourself (-c option) of all emails sent to your lab or course instructor when using the mail utility.
- Documentation (comments) for exercise programs is optional.

Sample terminal session:

```
[lee@bobby keys]$ more data4twelve
11
0.0 13.0 27.0 -27.0 -9.0 5.0 11.0 1.0 21.0 -1.0 16.0
[lee@bobby keys]$ g++ ex12.cpp
[lee@bobby keys]$ ./a.out
Lee Misch Lec# 10__ Lab# 10__ Exercise# 12
Please enter name of the input file
data4twelve
  Input Array    2nd Array
      0.00000      0.00000
     13.00000      2.35133
     27.00000     54.00000
    -27.00000     -3.00000
     -9.00000    -18.00000
      5.00000      1.70998
     11.00000     22.00000
      1.00000      1.00000
     21.00000     42.00000
     -1.00000     -1.00000
     16.00000     32.00000
Sum of all values in array 1: 57.00000
Sum of first 5 values in array 2: 35.35133
```

Make sure you test your program adequately.
Use the mail utility to send your program file to your lab instructor.

Return to exercises list