CS 135
Exercise #7
Point value: 40
Date due: email your source file (.cpp file) to your lab instructor by 11:59pm Sun, Mar 5

New Skills Practiced (Learning Goals)

- Problem solving and debugging.
- Linux redirection to read data from a file.
- Use of count-controlled loop.
- Nested loop.

An abundant number is an integer greater than 0 such that the sum of its proper divisors is greater than the integer. For example, 12 is abundant because 1+2+3+4+6 = 16 which is greater than 12.

A deficient number is an integer greater than 0 such that the sum of its proper divisors is less than the integer. For example, 8 is deficient because 1+2+4 = 7 which is less than 8.

A perfect number is an integer greater than 0 such that the sum of its proper divisors is equal to the integer. For example, 6 is perfect because 1+2+3 = 6.

Data File Description
A data file for this exercise will consist of several integers separated by whitespace. The first integer in the file will be greater than 0 and will indicate how many data values are to be processed.

Here are 2 sample data files:

```
3
17   -5    246

8    0    11    -12354    894
183   -67   14   33333
```

Design and implement a complete C++ program that

- displays your name, lecture and lab section #s, and exercise #
- using Linux redirection, reads the first integer in the file to determine how many data values to process
    - for each of the data values, determines if the integer is abundant, deficient, perfect or neither and
      displays a message that includes the integer and which of the 4 categories it falls into
    - if the integer is abundant, counts how many factors it has and displays a message that includes its factor count

When the program compiles and runs correctly, use the mail utility to email a copy of the program file to your lab instructor. Make sure the subject line of your email includes your name, lecture and lab section #s, and the exercise # if you wish to receive full credit.

NOTES:

- Make sure you choose enough test data to ensure that your program meets all the requirements.
- 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.

- A comment with your name, lecture section#, lab section#, and exercise# should be at the start of your program file.

Sample terminal session:

```
[lee@bobby keys]$ more data4seven
3
17   -5    246
[lee@bobby keys]$ g++ ex07.cpp
[lee@bobby keys]$ ./a.out < data4seven
Lee Misch Lec# 10__ Lab# 10__ Exercise #7
17 is deficient
-5 is not abundant, deficient or perfect.
246 is abundant and has 8 factors
```

Return to exercises list