

# Homework Assignment #2

**EE475**

**Fall 2004**

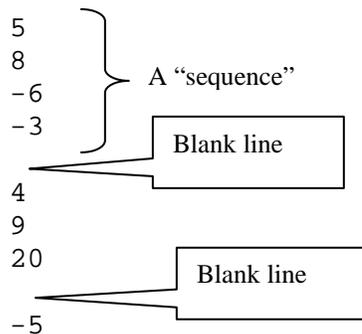
Assigned: Tuesday, September 21, 2004

Due AT THE START OF CLASS on Thursday, September 30, 2004

This assignment, like all homework assignments in this course, is to be completed individually.

You need to hand in a concisely written description of your approach, your actual source code including comments, the output of the program, and **comprehensive** work you did to check and verify the results.

A network-attached measurement device creates a DOS-format text file consisting of a sequence of integers, one per line, proportional to the temperature of items passing a sensor on a conveyor system. The text file may contain more than one sequence, which is indicated by a blank line separating one sequence from the next. Each sequence may have a different number of integers. For example:



Your task: write a C program that opens the text file, reads and interprets each line, and prints out a summary after each sequence is read. The summary must include:

- The *number* of measurements in the sequence
- The *average* of the measurements in the sequence
- The *maximum* and *minimum* value in each sequence

When the end of the file is reached your program must close the text file and print a "done" message.

Your program should either allow the user to give the text file name on the command line (if you know how to do that), or prompt the user to enter the name of the text file and read in the file name string. Your code must handle basic errors such as being unable to open the file, not finding any integers, and handling the presence or absence of blank lines at the start or end of the file.

An example test file will be provided on the course web site, and you should also make your own test files to show how your program handles various errors.

Incidentally, some of the file and string I/O routines in the standard C library are:

<code>fopen()</code>	<code>fgets()</code>	<code>fscanf()</code>
<code>fclose()</code>	<code>gets()</code>	<code>sscanf()</code>

Refer to a C language reference to see how these library functions operate.