CISC105 Spring 2007 Lab07

- Code incrementally, save and test often! I will save many versions of a long program on the way to its completion, so that if I make a big mistake or lose a file I can retreat to a previous, working version. **Programs that do not compile lose 50 percent** even if they "look" like they should work, so compile and test, compile and test.
- **START PROJECT TWO!** It is going to take longer than the first one.
- Every file that is opened with fopen() should be closed with fclose().
- Some programs below are associated with a question. Answer the questions using C comments below your code in the program file.

For each numbered problem below you will write a small program. Name each program lab07.n.c, where n is the number in the list below. For example, the name of the file for the first will be lab07.1.c

Programs

- 1. Copy and modify the example program from the class website called fcnReturnValues.c. Add statements to the program that will print the memory addresses of the variables used. Then **draw the stack** as it would appear after the assignment statement in the sum() function, the first time sum is called. Label the drawing with the real memory addresses printed by your code running on Strauss. This drawing should be on a separate piece of paper to be included with your script.
- 2. Create a file called data1.txt. Put ten temperature numbers in it, separated by spaces, and save it. Now write a program that reads the ten numbers into a single double variable with a loop and prints them out as they are read.
- 3. Modify 2 so that the same ten temperatures are read into a type double array of ten elements. After the array is full, use a **separate** loop to print the contents of the array.
- 4. In lab05.2.c you wrote a table of multiples of three. Modify this program so that it writes the data to a file using fprintf. You will have to use the mode "w" instead of "r" when you open the file with fopen. Be sure to use the name of a file that you want the data to go to, since you will write over that file.
- 5. Copy the previous program and modify it so that it prints to two different files. You will need to declare two different FILE * variables, and to initialize each one separately with fopen.

Print all the odd multiples of three to one file, and all the even numbers to the other. You'll have to add some newlines to your code to make the data look the same in the two files.

6. Generate and write 100 pseudo-random numbers to a file. Then, in the same program, read the numbers from the file you just wrote into an array. Print the array using a function that takes two parameters: the array (by reference), and the array size (by value).

You should have a total of 6 programs named lab07.01.c to lab07.6.c. Make a single script file (see lab00 for the scripting instructions) where you cat, compile, and run each one in its final form (if it didn't compile, don't run it in the script - mark the place in the printed script file with a colored marker so it stands out).

On the first page of every printed copy for this course, your name, section, and TA's name must appear.

Submit all program and script files on MyCourses before midnight Thursday of next week, and give the paper version to your TA at the beginning of your Friday lab (or in lecture Friday if you have a Wednesday lab). Note: cat, compile, and run each program in order! Do *not* cat all programs, then compile, etc.