CISC105 Spring 2006 Lab06

- Write a program for each of the following problems. Be sure to save every separate program. All programs must be properly commented and indented (see Assignment Standards on the class website).
- Be sure you spend time coding over Spring Break. You are learning a new language, and you could forget a lot in 9 days!
- Name each program lab06.n.c, where n is the number in the list below. For example, the name of the file for the first will be lab06.1.c. Put the files in your lab06 directory.

Programs

1. Copy the program lab06code.c from the website using the unix copy command and the same directory you used in lab02. Add print statements to every function that will print the addresses of the variables alpha and beta. Draw what the stack looks like, as shown in class, when the program is executing code inside the function **c()** (this will be one stack with multiple function calls on it.) Label the stack drawing with the names of the functions, names of the variables, and the memory addresses of the variables AS SHOWN IN YOUR SCRIPT FILE. Because addresses may be different when you run the program again, you may script this run separately so that you have sufficient time to work on labeling the drawing. Be sure that a script containing the numbers that match your drawing is attached to your assignment and uploaded.

2. Declare an array of integers, size 100, and fill it with the multiples of seven starting with zero (how?). Then print it out with a loop, using a newline every 10th element. Use the "-o" option in the compiler to give your executable the name "sevens". Use information from H&K section 2.6 so that all the columns are aligned and appear exactly as follows:

```
% ./sevens
   0   7  14  21  28  35  42  49  56  63
  70  77  84  91  98 105 112 119 126 133
 140 147 154 161 168 175 182 189 196 203
 210 217 224 231 238 245 252 259 266 273
 280 287 294 301 308 315 322 329 336 343
 350 357 364 371 378 385 392 399 406 413
 420 427 434 441 448 455 462 469 476 483
 490 497 504 511 518 525 532 539 546 553
 560 567 574 581 588 595 602 609 616 623
 630 637 644 651 658 665 672 679 686 693
%
```

3. 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 with a loop and prints them out.

4. Modify 3 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.

5. Place the multiples of seven generated by your program from 2 into a data file called sevens.txt. To do this, use the Unix redirection operator **>** like this:

   ```
   sevens > sevens.txt
   ```

   Now you will read them into two separate arrays. Make two integer arrays of size fifty. Start a loop reading the elements from the data file. As each element is read, print it out.

   Next, still inside the read loop, use *mod* so that elements numbered 0, 2, 4 etc. go into the first array, while elements numbered 1, 3, 5 etc go into the second array (Looking at the data, what would be good names for the arrays?).

   After all the data is read, print the arrays separately.

You should have a total of 5 programs named lab06.1.c to lab06.5.c. Make a single script file (see lab00 for the instructions) where you cat, compile, and run each one in its final form.

Submit all 5 program files *and* your script on WebCT, and give the paper version of the complete script file **only** on paper to your TA at the **beginning** of your next lab (Wednesday labs submit Friday at the **beginning** of class). Note: Cat, compile, and run each program in order! Do *not* cat all programs, then compile, etc.