CSCI-15 Lab 4, due 10/16/18

Please work in groups of two, unless I clear it with you directly.

Recall lab 3, the `makeSpiral()` lab. Your job is to make new versions of `makeSpiral()` and `printSpiral()` that take the base of a 2-dimensional array (an `int *`, not an `int [][]`), and the number of rows and columns in the actual 2-dimensional array, and which do the same tasks you did with the fixed-size array in lab 3. The difference here is that you will fill the entire array in the function, by its actual dimensions, not just a corner of a large array. If you did not make a `clearArray()` function to fill the arrays with zeroes, create one with the same parameters, otherwise adapt your function to take the pointer instead of the array. I know you can just initialize the arrays with all 0s, but make and use the function anyway.

The function headers will look something like this:

```
void makeSpiral( int *p, int nRows, int nCols )
```

Obviously, the print function will take a file handle, too. Here, nRows and nCols will be the actual dimensions of the arrays, not the dimensions of a corner. Use the matrices.cpp example as a guide to how to mimic array indexing off of the pointer (the base address). You will get the bases of the 2-d arrays with an expression like `&a[0][0]`. Hold the bases of the 2-d arrays in an array of `int *` to pass into your functions.

Write a program that declares a series of arrays, each no larger than 15 by 20, fills each with zeroes, calls the `makeSpiral()` function on each with its own dimensions, and then calls the `printSpiral()` function on each to print the spirals. You MAY NOT just use a single large array for this like you did in lab 3.

Again, you may use other functions if your design needs them, and all output must be directly to a text file. Pass the output file name to the program on the command line via `argc` and `argv[]`. Put the various function calls in a loop, not stretching them out separately in `main()`.

Create arrays of at least the following sizes, plus several of your own choice:

1 by 1, 2 by 2, 3 by 3, 4 by 4, 5 by 5, 4 by 7, 7 by 4, 4 by 8, 8 by 4, 15 by 20

You MAY NOT just declare a single 2-d array for the spirals; you MUST have multiple 2-d arrays of different dimensions.