C101 Homework #11 Due 11/28/12

This homework deals with working with arrays, specifically integer arrays. Write a program that will add two "Large Ints" together and then display the result on the screen along with the two inputted numbers. This program will only work on positive numbers, and we will assume a maximum number of digits per number to be 20, but you can easily test for larger numbers.

The idea here is to in a way create our own "Very big integer type" by using integer arrays. Right now the largest "short int" is 32,767 and the largest "long int" is 2,147,483,647, where the last has only ten digits. But what if we wanted an integer that had 20 digits or even more. We could use int arrays and create our own data type by letting each digit in our number be stored in each array element. We could actually create an integer that has more than 1,000 digits in it.

The way it works is to store each "Large int" in an integer array, so if the user enters the number "7834" and we have an int array named N, then N[0] has 7, N[1] has 8, N[2] has 3 and N[3] has the number 4 in it. So depending on the array size would determine the number of digits our "Large int" could be.

This program deals with having the user enter any positive large integer, lets use a maximum number of digits to be 20, and hence this will be our array size. The number will then have to be stored in an int array, but each separate digit of the number must be stored separately in the correct array element location. To extract one digit at a time one needs to read each digit first as a single character, convert this character to an integer and store this integer into the correct array location. Then read the next character digit and so on until the newline is encountered.

Basically you store each "Large int" in there own integer array. Each inputted number will be stored the same way, but in different arrays. Then the program will just add each separate element from each of the two arrays and store the result in another array. All three arrays will be of the same size and type; for the size we can use a global constant. The program should continue as long as the user chooses, example output below:

```
Program will add two large numbers

Please enter first number --> 8764
Please enter second number --> 98
00000000000000008764
00000000000000000098
------------------------------
00000000000000008862

Continue ‘Y’ or ‘N’ →
```

Note that the program does not need to show a plus sign, "+"; and you can leave the leading zeros, '0', out of the display if you want to. To receive full credit you must use at least three functions for this program, one function that will allow user to enter any large number, another one to add the two numbers and store in a third array, and the third function to display the results. Remember to keep the main function as simple to follow as possible. We will work on an algorithm in class. And make sure when testing data to test for some of the odd cases.

Hints -- Get program to work for simple cases first, i.e., entering number and storing into an array the way you wish it stored, then display array. Remember to do one small part at a time, get that to work then add another part. Very important to write a good algorithm, this will save you time.