**Searching and Sorting**

[Here](http://borax.truman.edu/180/lab11srchsrt/template.cpp) is the main function for a program that does the following:

1. Read a data file named data.txt that contains integers. [Here](http://borax.truman.edu/180/lab11srchsrt/data.txt) is a data file you can use, or you can make your own. The data file is guaranteed to have at least two and no more than 100 values.
2. Read the values into an array, keeping track of how many values there are.
3. Sort the array using your choice of sorting algorithm.
4. Copy the sorted array’s elements to a vector.
5. Repeatedly:
   1. Prompt the user for an integer value
   2. Search for the value in the array using an enhanced linear search (one that recognizes when the element will not be found because the current array element exceeds the value of the searched-for element). Output the position of the element if it is found, or output that the value was not found if it is not.
   3. Search for the value in the vector using binary search. Output the position of the element if it is found, or output that the value was not found if it is not. Obviously, the results of this search should be identical to those for the linear search in the array.
   4. End the program when the user enters the sentinel value 0.

The program produces output to screen that looks like this:

Read 47 values from data.txt

Enter a value to search for: 400

Value is in the array at position 31

Value is in the vector at position 31

Enter a value to search for: 123

Value was not found in the array

Value was not found in the vector

Enter a value to search for: -361

Value is in the array at position 13

Value is in the vector at position 13

Enter a value to search for: 37

Value was not found in the array

Value was not found in the vector

Enter a value to search for: 0

At a minimum you must implement the functions sort\_array, copy\_array\_to\_vector, linear\_search\_array, and binary\_search\_vector. You may write other functions if you wish. You cannot change the main function in any way.

You can copy as much code as you wish from the textbook (modified with the items noted below), from the sample programs provided by the instructor, and from the slides.

Remember that *every* variable that holds a size or an index (i.e., a position) must be declared as type size\_t.

Remember that you can *only* use square brackets to access array elements, but that you *never* use square brackets to access a vector element; instead, you use .at().

Remember that a vector is *never* passed by value. It is passed either by reference or by constant reference.

When you are satisfied with your program, by the due date of 5pm, Saturday, 2 November, submit it to the usual submission page.