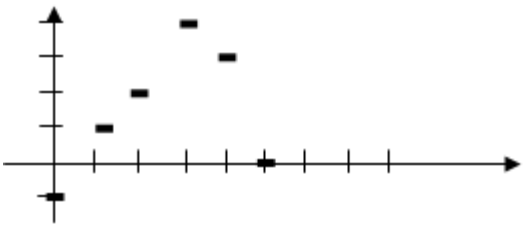
CSCI 58000
Program 1

Due 10/5/16

Consider the following problem. You have an array A[0], A[1], …, A[n - 1] of distinct integers that has the following property: The values in the array increase up to index p for some p, $0 \le p \le n-1$, and then decrease for all indices beyond p through position n - 1. You want to find the p index at which the peak value occurs, and what the peak value is.

Example: In the 6-element array illustrated below, the p index is 3, and the peak value is 4.



(10 pts) a. Describe [write an English paragraph, not code] a divide-and-conquer algorithm that can solve this problem.

(10 pts) b. Do a formal analysis (for which you can assume that n is a power of 2) to prove that your algorithm does $\Theta(\lg n)$ work units, where the work unit is comparison of array values.

Turn both your algorithm description and analysis in on paper **(for the TA)**.

(20 pts) c. Implement your algorithm in a C++ program called *peak.cpp*. [As you know, C++ arrays are 0-origin indexed, so don't write your code in a way that pretends the array is indexed 1 through n.] Recall your program will be run on Visual Studio 2013. If you use Visual Studio to create your program, ***be sure that you choose Win32 Console Application and start with an empty project***. I expect that, on all coding assignments, you will follow good coding practice – reasonable identifier names, modularization using functions, clear comments, etc.

Read the input values (i.e., the array values) as a series of integers, one per line, from a text file called *peak.txt* – make up your own file for testing your program (see the reminder on the C++ Resources page about reading data from a file). You can assume a maximum array size of 32. Turn in only *peak.cpp* (not all your project files or data files) via Canvas Assignments.