Lab 4 (29 Jan 2018)

Problem 1 [Maximum sum sub-array]: Given an array A of integers write a program to return indices (i,j) ($0 <= i < j <= n-1$) such that the sum $A[i]+A[i+1]+...+A[j]$ is maximum for all sub-arrays. For e.g. if A = [-2, 10, -4, 12, -9], then i=1,j=3 would give the maximal sum (10+ (-4) + 12=18). Your algorithm should run in O(nlogn) time.
Hint: Think of a divide-and-conquer strategy. You can implement a naive $O(n^2)$ algorithm to check the correctness of your divide-and-conquer algorithm. Do you think you can solve this problem in O(n) time?

Problem 2: [Min & Max with least comparisons] Given an array of integers write a program to print the minimum & the maximum element in the array with the *least* number of comparisons. For e.g. a naive approach would be to have a *min* and *max* variable and compare each element of the array with the current min and max, updating them if needed. The number of comparisions would be 2n here. (2 comparisons) x (n elements). Can you do better? By how much? Make your program print the total number of comparisons done for a given input array.

Problem 3*: Try coding the Divide-and-conquer selection algorithm discussed in class.