CSCI 421 Design and Analysis of Algorithms Spring 2020

Lecture 1 Activity 1

**Faster 3-Sum**

1. (50 points) As discussed in class, implement a comparison-based faster 3-Sum algorithm with running time O(n2). Run the program against the following input array (assuming it is already sorted). Attach your code and screenshots of the output here.

-25 -10 -7 -3 2 4 8 10

O(nsup2) =

for (int i = 0; i < n; i++)

for (int j = i+1; j < n; j++)

if(a[i] + a[j] == 0)

cnt++;

1. (50 points) As discussed in class, implement a hash table based faster 3-Sum algorithm with running time O(n2). Run the program against the following input array (note: it does not have to be sorted). Attach your code and screenshots of the output here. Hint: put the array values into a hash table.

-10 -7 -25 2 -3 4 10 8