1) (5 pts) ANL (Algorithm Analysis)

What is the best and worst case runtime for the following algorithm, in terms of the input parameter n? Give a brief explanation for your answers.

```
int foo(int * arr, int n){

   if (n == 0)
      return 0;

   int j = 0, i;

   for (i = 0; i < n; i++)
      if (arr[i] > arr[j])
         j = i;

   int nLen = n - j - 1;
   return arr[j] + foo(arr + j + 1, nLen);
}
```

Best Case

The for loop runs and sets j = n - 1, which means that nLen gets set to 0. In this case, the subsequent recursive call will immediately return 0 and the original recursive call will return the value of the last array element. The run time in this case is **O(n),** since the entirety of the execution includes one for loop that runs n times and a few other simple statements. From a conceptual standpoint, the for loop identifies the index in between 0 and n-1 that stores the largest value within that range.

Worst Case

The worst case is when the array is sorted in reverse order. Every call eliminates only 1 value at the cost of n operations. The total runtime becomes **$O(n^2)$.**

Grading: 2 pts for each answer, 1 pt for all of the explanation.