1) (10 pts) ANL (Algorithm Analysis)

Give the Big-O run-times for each of the following operations in terms of the variables given in the description. When a particular implementation is not explicitly stated, assume an efficient implementation is used. In order to earn full credit, you must provide a simplified, asymptotically tight bound. (For example, if O(n) was the correct answer, O(5n) and $O(n^2)$ would not receive full credit, even though both are technically correct.)

a) Merging a sorted array of size ***m*** with a sorted array of size ***n*** into one sorted array. **O(m+n)**

b) Creating a heap out of ***n*** unsorted integers. **O(n)**

c) **Worst case** run-time of running a Quick Sort on ***n*** integers. **$O(n^2)$**

d) Inserting an element to the front of a linked list with ***n*** elements. **O(1)**

e) Deleting ***m*** items, one by one, from an **AVL** tree which originally contains ***n*** items (***n*** ≥ ***m***) **O(mlg n)**

f) A sequence of ***p*** push operations onto a stack that originally had ***n*** elements on it. (Assume the stack has enough space to handle the sequence of push operations.) **O(p)**

g) **Average case** run time of an insertion sort on ***n*** unsorted integers. **$O(n^2)$**

h) Calculating a^b mod ***c***, using fast modular exponentiation, assuming that each multiply and each mod operation take O(1) time. **O(lg b)**

i) Pre-order traversal of a binary tree with height ***h*** and ***n*** nodes. **O(n)**

j) **Worst case** run-time for searching for an element in a binary search tree with ***n*** nodes. **O(n)**