Gauge Hartwell

assignment 2 write-up

1A)

for (0 to size)

this.insertAt(i + size, otherList.retreiveAt(i));

otherList.clear();

1B) Big O = O(n). This is because the program has to read in n lines from list B and add them on to list A. Reading the lines is n work, and adding them to the end of list A is constant work.

2A)

ArrayList<Address> temp;

for (0 to size)

if (this.state == itemToMatch.state)

temp.insertEnd(this address);

this.removeAt(i);

return temp;

2B) Big O = O(n). This is because the function reads through the whole list to check each item, then removes the item and adds it to the other list if it matches the search criteria.

4A) size = other.length; //used for length of for loop

node\* temp = other.head;

for (0 to size)

this.insertEnd(temp);

other.head = other.head.next;

delete temp;

temp = other.head;

other.length--;

4B) Big O = O(n). N to run through linked list, constant time for insertEnd function.

5A)AddressLinkedList temp;

size = this.length;

for (0 to size – 1)

if (this.state == itemToMatch.state)

temp.insertEnd(this);

length--;

return temp;

5B)Big O = O(n). N to check each item in list, constant for insertEnd function