James Cooper | Lab 6 - Running Time Analysis

1:

n = manyItems

First Line 3 is a if so 1 operation.

Then one operation to declare biggerArray[] as integer array and inizialize the size of it.

The main part of this method is Lines 7 and 8.

It contains a loop, which in the worse case runs n times

In each iteration there are 3 operations: increment i by 1, compare i with manyItems or n, and assign data[i] to biggerData[i]

The worst case of this is 3n

Next is 3 more operations, one to assign data with biggerArray, another to set data[manyItems] to element, and finally one to increment manyItems by 1.

Adding this together we get 3n + 1 + 3 = 3n + 4. In Big-O its time complexity is O(n).

2:

n = manyItems

First the rest of the operations other than lines 6 - 8 are constant and will not add to the time complexity of the method in relation to n, but.

Line 3 = 1, 5 = 1, 9 = 1 with a total of 3 outside the main bulk of the method that will actually take time.

Onto that, lines 6 - 8 has a total of 4n, with index < manyItems = 1, increment index by 1 = 1, checking if target equals data[index] = 1, and incrementing answer by 1 = 1. So those 4 run a total of n times, hence 4n.

So, 4n + 3 in Big-O its time complexity is O(n).

3:

n = how many nodes are in the list, from head to null

Say the list has 5 nodes, n = 5

Outside the for loop everything totals to 5 operations.

1 on line 5, 1 on line 6, 1 on line 7, 1 on line 8, and 1 on line 10

The for loop will run 4 operations everytime it is ran.

i < position = 1 operation, cursor != null = 1 operation, increment i by 1 = 1 operation, and cursor = cursor.link = 1 operation.

Therefore 4n is the loop time complexity and

4n + 5 in Big-O notation is O(n).

4:

n = how many nodes are in the list, from head to null

Say the list has 7 nodes, n = 7

Outside the main loop there are 4 perations

IntNode cursor = null;

int answer = 0;

return answer;

and cursor = head;

The for loop will run 3 operations each iteration

cursor != null;

cursor = cursor.link;

and answer++;

Therefore 3n is worst case for the loop

3n + 4 in Big-O notation is O(n)