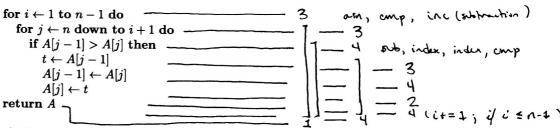
1. For the following pseudo-code, count the primitive operations considering the best and worst case. What conditions would lead to the best case? What conditions would lead to the worst case?

Algorithm BubbleSort(A):

Input: An array, A, of n comparable elements, indexed from 1 to nOutput: An ordering of A so that its elements are in nondecreasing order



a) Use operation counting to determing the running time in terms of input size n, T(n), for the algorithm above:

T(n): $\frac{12}{3}n^2 - \frac{3}{3}n - \frac{3}{3}$ (worst case) — Occors when the array is in reverue sorted order.

T(n): $\frac{4n^2+3n-3}{2}$ (best case) - occurs when array is already sorbed.

$$T(n): 3 + \sum_{i=1}^{n-1} [3 + (n-i)[4 + (3+4+2)+4] + 4] + 1$$

for the interconstruction interconstruc

b) What is the Big-Oh time complexity of this algorithm? Set up and solve the inequality to verify your answer.

BubbleSort is in $O(\underline{\qquad \qquad \qquad })$

$$\frac{17}{2}n^2 - \frac{3}{2}n - 3 = cN^2$$
; let $c = \frac{17}{2}$, $\frac{3}{2}n^2 - \frac{3}{2}n - 3 = \frac{17}{2}n^2$.

This inequally halds for all $n \ge 0$.

| 2. | For the following pseudo-code, count the primitive operations considering |
|----|---|
| | the best and worst case. What conditions would lead to the best case? |
| | What conditions would lead to the worst case? |

Algorithm LinearSearch(A, n, s):

Input: An array, A, of n integers, indexed from 0 to n-1 and an integer s which is the integer we are searching for

Output: The index of the first element that matches s, or -1 if no element matches s

a) Use operation counting to determing the running time in terms of input size n, T(n), for the algorithm above:

$$T(n)$$
: 4+60 (worst case) - item is not in the list $T(n)$: 6 (best case) - item is first in the list $T(n)$: 3+ $n(2+4)$ + 1

b) What is the Big-Oh time complexity of this algorithm? Set up and solve the inequality to verify your answer.

LinearSearch is in $O(\underline{\hspace{1cm}}$

3. The following pseudo-code swaps even indexed elements of an array with their consecutive odd indexed elements (e.g element at index 2 with element at index 3, element at index 4 with element at index 5 and so on). Count the primitive operations considering the best and worst case. What conditions would lead to the best case? What conditions would lead to the worst case?

Algorithm Swap(A,n)

Input: Array A containing integer numbers and integer n which shows the length of the array

Output: Array A which its even indexed elements are swapped with their consecutive odd indexed elements

$$T(n)$$
: $3+11\frac{\alpha}{2}$ (worst case) } thur on the same because the objection $T(n)$: $3+11\frac{\alpha}{2}$ (best case) } thur on the same spectrum, regardless of the input every; elements, $T(n)$: $1+1+\frac{\alpha}{2}(2+4+3+2)+1$

b) What is the Big-Oh time complexity of this algorithm? Set up and solve the inequality to verify your answer.

Swap is in $O(\underline{\qquad} \land \underline{\qquad})$

* This algorithm has an error when n is odd; the loop exember and others to assect the ir2 element, but this will be out along if i = n-1.