CS2223 D Term 2020 Quiz 3

(1 point) Question 1:

I pledge that I am taking this quiz on my own, with help from no one else and no notes: "My brain is open. . . ."

(3 points) Question 2: SelectionSort and $\Theta(n^2)$
SelectionSort is $\Theta(n^2)$. Why?

A.) SelectionSort makes the same number of comparisons regardless of the arrangement of the input list.
B.) SelectionSort makes fewer comparisons if the input list is already sorted.
C.) SelectionSort makes more comparisons if the input list is already sorted.
D.) SelectionSort makes fewer comparisons if the input list is in *reverse* sorted order.
E.) SelectionSort makes more comparisons if the input list is in *reverse* sorted order.

(3 points) Question 3: SelectionSort v. BubbleSort
SelectionSort is $\Theta(n^2)$, but our implementation of BubbleSort is not. Why not?

A.) Our implementation of BubbleSort is $O(n^2)$.
B.) Our implementation of BubbleSort is $\Omega(n^2)$.
C.) Our implementation of BubbleSort is $\Omega(n)$.
D.) A) and B)
E.) A) and C)

(3 points) Question 4: A sorting algorithm is called *stable* if it. . .

A.) . . . has constant time order of growth.
B.) . . . has the same worst-case running time as best-case running time.
C.) . . . always sorts duplicated elements in the same order that they appear in the input.
D.) . . . permits data to be sorted in-place.
E.) . . . employs the operoater $>\approx$, known as "greater than or equus".

(1 point) Bonus Question: Comparisons versus Swaps

SelectionSort and BubbleSort both make $O(n^2)$ comparisons, but SelectionSort makes only $O(n)$ swaps while BubbleSort's makes $O(n^2)$ swaps because:

A.) Best case for BubbleSort is an already-sorted list.
B.) Worst case for BubbleSort is a reverse-sorted list.
C.) SelectionSort makes at most one swap per pass on its list.
D.) Best case and worst case for SelectionSort are the same, i.e. it is $\Theta(n^2)$.
E.) SelectionSort and BubbleSort might not make the same number of swaps.

NOTE: All of these are true, but only one is the BEST answer to the question.