Assume that an array consists of at least 10 elements. The worst case time (in the big-oh sense) occurs in linear search algorithm when**last element,middle of array,not in array.** Given an array-based list implementation, deleting the current element takes how long in the average case**O(n)** There are some algorithms that do not terminate for certain inputs.**false**T(n)=n^2**8n**Hardware vendor XYZ Corp**10n**Which of these is the best upper bound for a growth rate of 5n+3?**O(1)** The Sequential Search algorithm is in O(n^2)**T**For all the algorithms for which we properly understand the running time analysis, the upper bound and lower bound will always be the same.**T**The lower bound for the cost of sequential search is Ω(1) since this is the running time of the algorithm in the best case.**F**The worst case for the sequencial search algorithm occurs when the array size tends to infinity**F**Which of these is the best definition for a stable sorting algorithm**Does not change**Which of these will NOT affect the RELATIVE running times for two sorting algorithms**CPU**The upper bound and lower bounds of the sequential search algorithm is in O(n) and Ω(n)respectively**F**In which case might the number of comparisons NOT be a good representation of the cost for a sorting algorithm**comparing**Insertion Sort (as discussed in class) is a stable sorting algorithm. Recall that a stable sorting algorithm maintains the relative order of records with equal keys.**T**What is the average-case time for Insertion Sort to sort an array of n records**O(n^2)** When implementing Insertion Sort, a binary search could be used to locate the position within the first i−1 records of the array into which record i should be inserted. Using binary search will:**Not speed up**In which cases are the growth rates the same for Insertion Sort**Worst and average only**The order of the input records has what impact on the number of comparisons required by Insertion Sort**Big Difference**When is Insertion Sort a good choice for sorting an array**Few records**If I is the number of inversions in an input array of n records, then Insertion Sort will run in what time**O(n+I)** The order of the input records has what impact on the number of comparisons required by Bubble Sort**None**What is the running time of Bubble Sort (as the algorithm is presented in this module) when the input is an array that has already been sorted**O(n^2)** In which cases are the time complexities the same for Selection Sort**Worst,Average,Best**The average number of inversions in an array of n records is n(n−1)/4. This is:**O(n^2)** An inversion is:**When a record**If I is the number of inversions in an input array of n records, then {Insertion|Bubble} Sort will require how many swaps**I**Consider an array A of n records each with a unique key value, and A​R​​ the same array in reverse order. There are a certain number of pairs, where an arbitrary position i and position jis a pair. Between these two arrays, exactly half of these pairs must be inverted**F**How many ways can n distinct values be arranged**N!** Which is the divide-by-twos increment series for an array of 23 elements**16,8,4,2,1**You must merge 2 sorted lists of size m and n, respectively. The number of comparisons needed in the worst case by the merge algorithm will be**M+n-1**In the worst case, the total number of comparisons for Mergesort is closest to**N log n**When is Mergesort a good choice for sorting an array**Fast algorithim**What is the average-case time for Mergesort to sort an array of n records**O(nlogn)** When is Quicksort a good choice for sorting an array**In most standard**Quicksort (as discussed in class along with pseudo-code) is a stable sorting algorithm. Recall that a stable sorting algorithm maintains the relative order of records with equal keys.**F**In which cases are the time complexities the same for Quicksort**Best,Average only**What is the worst-case cost for Quicksort to sort an array of n elements**O(n^2)** The lower bound for a problem is defined to be**Best possible**What is the worst-case cost for Quicksort's partition step**O(n)**

**Which statement is false**Every binary tree has at least one node**Which is the best definition for collision in a hash table?** Two records with different keys have the same hash value