

1. What is the worst-case time complexity of each of the following algorithms? (a) Quicksort, (b) Mergesort, (c) Insertion sort, (d) Heapsort, (e) Bubble sort. (10 points, 2 each)
2. Continuing on Problem 1, what is the average-case time complexity of each algorithm? (10 points)
3. Give a stack with 3 entries. Assume that initially the stack is empty. Now we want to use a sequence of push and pop operations on 5 objects A, B, C, D and E. The 5 objects will enter the stack in the order of A 1st, B 2nd, C 3rd, D 4th, and E 5th. However any object can be popped out of the stack whenever it is on the top of the stack. For example if the 5 objects are popped after they are all pushed into the stack, then the output sequence will be EDCBA. On the other hand, if each object is popped right after it is pushed into the stack, then the output sequence is ABCDE. Which sequence(s) in the following can be a valid output sequence? (a) DECBA, (b) CDAEB, (c) BDCEA, (d) BCAED, (e) ADECB. (10 points. You must answer yes or no for each choice independently. You get 2 points for each correct choice and minus 2 for each incorrect choice.)
4. Continuing on Problem 3, can you give a general rule to check whether a sequence is a valid output sequence or not? (10 points)
5. How many different binary search trees can have three nodes with values of 1, 2 and 3, respectively? Similarly how many can have 4 nodes with values of 1, 2, 3 and 4, respectively? (15 points)
6. What is a priority queue? Describe how to use an ordered list to implement a priority queue. Also describe how to execute the various basic operations on the queue. (15 points)
7. Prove that a comparison-based sorting algorithm has a worst-case time complexity of $O(n \log n)$. (15 points)
8. The Tower of Hanoi problem can be easily solved by a recursive algorithm. Give such an algorithm in a pseudo-code format. Now if you are asked to implement a non-recursive (or iterative) algorithm to solve the Tower of Hanoi problem, how will you proceed? (15 points)