

Problem 7: Problems on Sorting and Related Concepts

Data Structures Lab (CS111)

In this practice problem, you need to write C programs to perform the following tasks:

1. You need to generate 1000 integer elements randomly (using *rand()* function) and store those elements in an array *A[]*. You need to take the upper limit on the the integers from the user. For example if the user provides you upper limit 2000, you need to generate 1000 integers between 0 to 2000. **[2 marks]**
2. Implement *quick sort* and *insertion sort* algorithms separately to sort *A[]* and store the sorted sequence in another array *B[]*. Count the number of comparisons and swaps are occurring in each case to sort *A[]*. Add 10 randomly generated number within the limit provides by the user in question 1 at the end of array *B[]*. Run *quick sort* and *insertion sort* algorithm separately to sort *B*. Again count the number of comparisons and swaps. **[12 marks]**
3. Write a modified sorting algorithm where you start sorting any array of size ≥ 50 using *quick sort* algorithm. In each iteration, the *quick sort* algorithm partitions the array in smaller sizes. When the array size of a partition becomes < 50 , use *insertion sort* algorithm to sort that partition. Use this modified sorting algorithm to sort *A[]* and *B[]*. Again count the number of comparisons and swaps. **[6 marks]**
4. Consider a sequence *C[]* where each element is either a digit in [0 to 9] or a capital letter in [A to Z]. Such a sequence is given as a user input. Write a program to rearrange *C[]* in a way that the digits occur first in the rearranged sequence followed by the capital letters. You need to do this rearrangement by traversing the whole array only once. **[5 marks]**