
Department of Computer Science and Engineering
Indian Institute of Technology Jodhpur
CSL2020 - Data Structures and Algorithms

March 7, 2022

# Lab - 8

1. Quicksort provides optimal performance in terms of both time as well as space complexity, however, it uses recursive function calls, which is expensive. In this exercise we will use Insertion sort to reduce the number of recursive function calls. Insertion sort is a reasonable choice for small size inputs. Your job is to implement a modified Quicksort in **Python** to sort an array of positive integers in ascending order. The modified Quicksort is same as Quicksort except that it calls a function applying Insertion sort on the partitioned subarray if the size of subarray becomes smaller than a threshold **T**. While applying Quicksort, you should always choose the final misplaced element of the array/subarray as the pivot. Your program will receive input in the following format.

   **N T m1 m2 m3 m4 m5 ... mN**

   where the first number **N** tells the length of sequence **m1 m2 m3 m4 m5 ... mN** and **T** is the threshold.

   **Print the content of array after each function call (including recursive calls) as output.**