

### Assignment 3: Sorting

- Due by: Friday 29th October, 2021 by 5pm IST.
- To be submitted to the following email address: office.of.gr@gmail.com
- The subject of the email should be: Assignment Number [3]: Algorithms, 2021
- Please clearly mention your name and roll number.
- Submit your work as a single pdf file. Additional material, code, etc can/should also be submitted, but there should be atleast 1 pdf, which has the entire assignment.
- Wherever there is code, in the assignments, the code should be well documented and easy to understand / follow.

### Questions

1. Show that the running time of the merge-sort algorithm on  $n$ -element sequence is  $O(n \log n)$ , even when  $n$  is not a power of 2.
2. Consider a modification of the deterministic version of the quick-sort algorithm where we choose the element at index  $\lfloor n/2 \rfloor$  as our pivot. Describe the kind of sequence that would cause this version of quick-sort to run in  $\Omega(n^2)$  time.
3. Describe and analyze an efficient method for removing all duplicates from a collection  $A$  of  $n$  elements.
4. Show that quick-sort's best-case running time is  $\Omega(n \log n)$ .
5. Implement in python, a bottom-up merge-sort for a collection of items by placing each item in its own queue, and then repeatedly merging pairs of queues until all items are sorted within a single queue.