Algorithms Analysis and Design: 2620301
Chongqing University of Technology — Oct 20, 2019
Lecturer Gou Guanglei — Handout 2

Problem Set 2

This problem set is due **Next Sunday at 11:59AM**.
Solutions should be turned in PDF form using LaTeX or word document.

Exercises are for extra practice and should not be turned in.
Exercises:

- Implement Quick-sort.
- Implement Selection.
- Implement Random-Quick-sort.

---

1. **(25 points)** Given a sorted array of distinct integers $A[1, ..., n]$, you want to find out whether there is an index i for which $A[i] = i$. Given a divide and conquer algorithm that runs in time $O(lgn)$

2. **(25 points)** Consider the task of searching a sorted array $A[1, ..., n]$ for given element x: a task we usually perform by binary search in time $O(lgn)$. Show that any algorithm that accesses the array only via comparisons (that is, by asking questions of the form "is $A[i] \leq z$?", must take $\Omega(lgn)$ steps.

3. **(25 points)** A *k-way merge operation.* . Suppose you have k sorted arrays, each with n elements, and you want to combine them into a single sorted array of kn elements.

   (a) Here's one strategy: merge the first two arrays, then merge in the third, then merge in the fourth, and so on. What is the time complexity of this algorithm, in terms of k and n?

   (b) Give a more efficient solution to this problem, using divide and conquer.

4. **(25 points)** Given the following list of numbers [1, 20, 11, 5, 2, 9, 16, 14, 13, 19], what would be the third pivot value using quicksort with *median of three method* (which chooses the median value of the first, the middle and the last element of the list as pivot) T his will be us eful when the original list is somewhat sorted to begin with.