CS21003: Algorithms-I (Theory)
Tutorial – 6 (Heaps and Hashing)
Date: 19-March-2020

1. Given k number of sorted (ascending ordered) arrays each having $\frac{n}{k}$ elements in it, your task is to merge all these arrays to form a n-element final sorted array (in ascending order).

   (i) Propose a simple solution to the problem which may run in $\Theta(n \log_2 n)$- time.

   (ii) Can you propose an efficient algorithm to solve the problem? What is the time complexity of your proposed solution? [Hint: Use Heaps]

2. You have seen binary heaps having at most two children for any node. Your task is now to develop a k-ary max-heap, that is each node is allowed to have at most k children. However, the max-heap property is to be maintained, i.e., every parent has higher values than any of its child. Develop algorithms to perform `heapify`, `insert`, and `deleteMax` opertions. Also, analyze the time complexity of these operations.

3. Given an unordered n-element array of integers and a number x, your task is to find a pair of element in the array whose sum is x. Give an efficient approach to solve this problem. What is your time-complexity and additional space required? [Hint: Use Hashing]

4. A hash function h is defined as, $h(key) = key \,\%\, 7$, with linear probing used to resolve collision. Given a hash-table size of 7, indicate the locations of the following inserted keys: 44, 45, 79, 55, 91, 18, 63, in the hash-table. What happens if quadratic probing technique is followed to resolve collision?

5. Which one of the following hash functions on integers will distribute keys most uniformly over 10 buckets numbered 0 to 9 for i ranging from 0 to 2020?

   (a) $h(i) = i^2 \; mod \; 10$

   (b) $h(i) = i^3 \; mod \; 10$

   (c) $h(i) = 11.i^2 \; mod \; 10$

   (d) $h(i) = 12.i \; mod \; 10$