**SSD**

**QUESTION-2:**

[**https://iiitaphyd-my.sharepoint.com/:v:/g/personal/mahesh\_dudhanale\_students\_iiit\_ac\_in/EUNupr8iUCtFvYiIQAtqJQwBXzFTXV8D71JgRouwzP3Cdg?e=X8eQg1**](https://iiitaphyd-my.sharepoint.com/:v:/g/personal/mahesh_dudhanale_students_iiit_ac_in/EUNupr8iUCtFvYiIQAtqJQwBXzFTXV8D71JgRouwzP3Cdg?e=X8eQg1)

**We cannot use iterator library**

**We can use to\_string method**

**Question: 1**

**Resources**

* Maintain an extra count variable (data structure augmentation) for duplicates
* So while deleting duplicates, whole node will be deleted without any issue
* See how to print trees efficiently

**Points by seniors**

* Does delete operation removes all occurrences of given value?
  + Yes
* In Question 1, to find K-th largest element can the time complexity be O(logN + K) ?
  + In ques1 it should be o(logn)

**Question: 2**

**Resources**

**Points by seniors**

* Dynamic hashing?
* map[key] – returns the value mapped to key
  + **Overload** the [] operator
* Take large prime number as hash table size (so collisions would be minimum)
* Convert all data types to strings and use rabin karp?

**Question: 3**

**Resources**

<https://www.codeproject.com/Articles/5425/An-In-Depth-Study-of-the-STL-Deque-Container>

**Points by seniors**

* deque () means that capacity will always be 0 and no insert/ pop is valid?
  + It will resize when insert
* No max size for deque ()
* We can use the hash map in question 2 for question 3
* deque[key] – returns the value mapped to key
  + **Overload** the [] operator
* When resizing the array,