**DS Project Assignment 1: Multi-Queue with priority**

**MCA 1st**

**1st SHIFT 2nd sem**

**GROUP MEMBERS**

* PIYUSH KUMAR (40714004418)
* SAHIL TANEJA (40214004418)
* NITIN SAINI (025)

**DS Project Assignment 1: Multi-Queue with priority**

Mode: Written Algorithm and program with output in printed format

Team: 2-3 members in a team

**Submission: 11/02/2019**

**Objective: Design an algorithm for creating a priority queue implemented using multi-queue on a dynamically allocated space.**

**Specifications:**

* Write an algorithm for priority queue implemented using multi-queue on linked list
* Removing of elements from a queue always starts with top priority queue. Elements are deleted as follows
  + Three elements from priority level 1
  + Then, two elements from priority level 2
  + Then, one element from priority level 3
* For adding elements in a queue, make sure that element should be inserted as per priority level with following consideration:
  + Element with priority level 1 can be added to priority level 1, but if level 2 queue is empty and level 1 queue shows ‘wait’ (i.e. more than 3 elements) then element should be added to level 2 and so on.

**DISCRIPTION OF Multi-Queue with priority**

**INSERTION**

**A node with a fixed priority between [1-3] will be inserted with the data associated with it.**

**Case 1**

If the level corresponding to that priority doesn’t exist, create a new level with the same priority and add the node with the data into it(enqueue)

And update count to the corresponding node. (where count is used to maintain the total number of nodes inserted.)

**Case 2**

if the level corresponding to the priority already exist then append the data to that level

And update count to the corresponding node.

**SUBCASE**

If count is greater than the required total number of elements (ex 3 nodes for 1 priority level) and another node of the same priority comes

* then we check the next available priority level if it doesn’t exist create one and insert the node to it
* if the level exists then insert the node to the level.
* Update the count corresponding to the level the node inserted.

**DELETION**

**Case 1:** if start\_level(which is a pointer to the very first level of the multiqueue it may be 1,2 or 3) is empty then show “**Multi-Queue with priority is empty**”.

**Case 2**: if the pointer of the queue of the level which is pointing to the list of inserted data is empty then next the level.

**Case 3**: Delete the first element of the queue.

**SHOW**

Pointer to the very first level must be traversed and within each level a temporary pointer is used to traverse the input data within the levels.

As soon as soon the temporary pointer of the level points to the NULL switch to the next level and this continues until the level pointer is NULL.