CL-210
Data Structures

Objectives:

- Queue Operations
- Priority Queue

Note: Carefully read the following instructions (*Each instruction contains a weightage*)

1. There must be a block of comments at start of every question's code by students; the block should contain brief description about functionality of code.
2. Comment on every function and about its functionality.
3. Mention comments where necessary such as comments with variables, loop, classes etc to increase code understandability.
4. Use understandable name of variables. 5. Proper indentation of code is essential.
6. Write a code in C++ language.
7. Make a Microsoft Word file and paste all of your C++ code with all possible screenshots of every task **outputs in Microsoft Word and submit word file. Do not submit .cpp file.**
8. First think about statement problems and then write/draw your logic on copy.
9. After copy pencil work, code the problem statement on MS Studio C++ compiler.
10. At the end when you done your tasks, attached C++ created files in MS word file and make your submission on Google Classroom. (Make sure your submission is completed).
11. Please submit your file in this format **19F1234_L8**.
12. Do not submit your assignment after deadline. Late and email submission is not accepted.
13. Do not copy code from any source otherwise you will be penalized with negative marks.

Problem: 1 |

Create a function that will duplicate the nodes of linked list depending on number of nodes. For example if the nodes are 3 it is to be duplicated 3 more time
Input: 3->4->5
Output: 3->3->3->4->4->4->4->5->5->5->5->5 You must use queue

Problem: 2 |Student Slip System [Priority QUEUE]

Suppose you have to manage the student slip issuance system using QUEUE & PRIORITY QUEUE data structure. There are 2 kinds of students those are waiting in the queue such as:

1- Student with Payment slip in their hands
2- Students without Payment slip

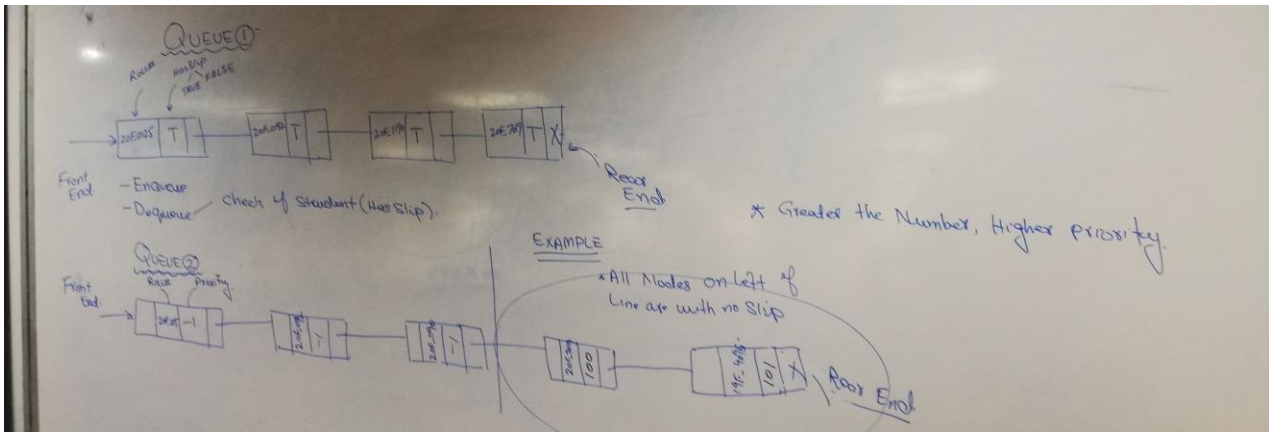
You have to perform the following operations.

1- Fill the QUEUE 1 by taking input from user (3 parts of LINKED LIST ROLL#, HasSlip, Linked to next node)

2- DEQUEU the element from QUEUE 1 and check if the student has slip

3- ENQUEUE the element in QUEUE 2 (which should be Doubly Linked list Priority QUEUE) with following data parts (ROLL#, PRIORITY (NO SLIP = -1, HAS SLIP = 1, 2, 3, 4,)). The higher the number the higher priority.

Use following Diagram to understand the procedure.


QUEUE (1)
Front End
- Enqueue
- Dequeue
Check if Student (Has Slip).
Rear End
* Greater the Number, Higher priority.
EXAMPLE
QUEUE (2)
Front End
*All Nodes on left of Line are with no Slip
Rear End