CSL 214 : Data Structures and Program Design II
H/W Assignment-1

1. Announcement Date: **23-03-2022**

2. Due date: Submit online by **11:59 PM on Wednesday 30th March, 2022**

3. The assignment has to be done individually.

4. For this assignment, Linked Lists are the preferred data structure. Hence, choose linked list (any type that you are comfortable with, singly linked list or circular linked lists or doubly linked lists) for any storage requirement you may have.

5. No Copying / sharing of code is allowed for this assignment. If any such case is identified, the original author and person who has copied, both will be penalised equally and zero marks will be awarded.

You need to submit your source files by attaching them to a mail and sending it on **dspd.assignment@gmail.com** by the common deadline. Please attach .c and/or .h and/or .txt files only. No .obj or.exe files should be attached. The subject should be marked as DSPD2-HW-Assignment-1: Your enrolment no.

Problem for R4 Batch (Enrolment Numbers BT20CSE091 to BT20CSE128, + ex-students):

Implement the following system using a linked list of structures where every node of that list represents a record of passenger name, passenger id, boarding train, boarding station, travelling class(Sleeper, 3AC, 2AC, 1AC), destination station, train id, Seat number (bogie number/seat number), and any other field you think that would be useful to passengers. You can also take a confirmation from the passenger whether upgrade of travel class is desired.

The passenger id can be thought as a key in the list and will represent a unique record in the list. The records should be always kept sorted according to the train id so that passengers boarding the same train have their data together.

You can assume number of bogies of each class and number of seats in each bogie.

Write the following functions :

-insert

- Insert a list of passengers and their details for the reservation
- I/p parameters: Reservation request that includes a list of passenger names, passenger ids, boarding train, boarding station, travelling class(Sleeper, 3AC, 2AC, 1AC), destination station, train id
- O/P: Reservation done successfully, partially or the reservation failed.
- Note – The set of passengers in a single reservation request should be allocated seats together. If all of them cannot get the seats together, then they need to be accommodated as close to each other in trains, that is, their bogie/seat numbers should be as close to each other.

-delete

- Deletes an element if the passenger cancels the reservation.
- I/p parameters: deleting all records of that particular passenger id.
- O/p: If node gets deleted print Reservation cancelled successfully or if it gets failed then print Reservation Cancellation failed.

-getListDestination

- Get the list of passengers having the same destination station and same train id

-SortByTravelDate

- Input – Passenger id
- Output – Display the list of destination stations for a particular passenger as per the dates of the travel.

-SortTrains

- I/p parameters: The train database that is the linked list with passenger data as given
- O/p: Display the train number and the travel date in the sorted order of number of passengers on the train.

- PromotePassengers

- Input – Train id and date of travel
- Output – For all the passengers on the train with train id and a particular date, passengers can be promoted to next travel class (Sleeper -> 3AC -> 2AC ->1AC) if seats are available. Passengers who had given consent for promotion to next class are considered in the order of their date of booking. Note that if 2AC passenger is promoted to 1AC, his/her 2AC seat becomes vacant and can be occupied by another passenger from lower class under promotion.