CSL 214 : Data Structures and Program Design II
Lab Assignment-1

1. Announcement Date: **25-02-2020**

2. Due date: Submit online by **12.00 noon on Wednesday 11th March, 2020**

3. The assignment has to be done either individually or a group of 2 students. In case of group formation, both students need to be from the same batch. All students of the same batch will do exactly the same problem. No late submission will be allowed in any case for this particular assignment. Early submissions are welcome.

4. 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.

5. 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. Though viva will be at different times for different students, the submission deadline is same for all. Please attach .txt, .c and .h files only. No .obj or.exe files should be attached. The subject should be marked as DSPD-2 Assignment-1: Your enrolment no.

Submission viva will be during practical hours starting from 11th March. For early submission any time with mutual consent of teacher and student as per availability of both is fine.

Problem for R5 Batch:

An apartment has 3 wings (wing-A, wing-B, wing C) 10 floor each (9 residential floors and ground floor is dedicated for parking vehicles). On each floor there are 4 flats (building capacity: 3*9*4=108).It has three parking levels such that Level A dedicated for wing –A , Level B for wing –B and so on. Each parking level has multiple compact and large spots.

Parking Condition: Two-wheelers can be parked in any spots. Four-wheelers can be parked in one large spot or two consecutive compact spots.

1. Initial list creation
   - Create linked list **Parking_Lot**to store parking spots information.(node info: Parking_level,dedicated_wing,no_of_compact_spots,no_of_large_spots,status_compactspots(array),status_large_spots(array)
     Note: Parking status 0- free, 1- allocated (initially status will contain 0).
   - Create linked list **Flat_Details** to store floor and wings details.(node info: wing_id, flat_id, no_of_two_wheelers, no_of_four_wheelers)
     Note:store vehicle details along with their vehicle numbers.
2. Write a function **Map_Flat_VehicleNo_Parkingspot** which will map flat in each wing to parking spots based on information in Flat_details and should follow the parking condition given. If any flat occupant cannot get parking spots in their respective parking level then remaining parking levels in order (i.e. if A is full then first B is searched and then C) should be taken into consider. **If any occupant has more than 1 four wheeler then only one large spot should be allocated in parking level dedicated to wing and others should be allocated other than dedicated parking level**. This function will map permanent parking spot to flat but their status will not be allocated all time.
3. Write a function **Park_vehicle**which will allocate mapped parking spot to vehicle just by reading vehicle number.
   Note:
   - If the vehicle number is not mapped with any parking spot then it is considered as guest vehicle. In such cases any free spot depending on vehicle type will be allocated.
   - For mapped vehicle if the spot is not free, then first free spot as per **Map_Flat_VehicleNo_Parkingspot** can be allocated.
   - This function should keep track on date, time, insiders and visitors, vehicle number for visitors.

Write a function **Display_Visitors** to get all visitors on particular date along with their vehicle type, number and the parking lot allocated.