30/11/20

DS ASSIGNMENT 2

ALLOTING SLOTS TO AIRLINES FOR TAKE-OFF & LANDING

(I) PROBLEM ANALYSIS AND SPCIFICATION

SLOT ALLOTMENT

- Get requests.
  - Request for arrival
  - Request for departure
- Prioritize
  - Have records in sorted order
  - search for specific airline
- Modify.
  - Modify and display slot order.

(II) ABSTRACT DATA TYPES

Data: Records - Text string airl_name
Integer frequency.

Operation: ~~buildt~~ sort_order() - To sort according to frequency.
compare() - To search for particular airline.

Data: Requests - Text string airline
Text string req (input: a/d)

Operation: modify() - To append the slot arrangement
display() - To display updated order.

III SOLUTION DESIGN

For the process to be handled by the software in the given scenario, addition of new values to existing data structure is key. This encompasses all cases of addition of a new element; adding at beginning, adding at end, adding after a certain element. Another point of importance is comparison for prioritising. For this process searching for an element in a data structure should have low time complexity.

To fufil the above requirements we can have; records stored in AVL tree with node as,

```
struct node{
        int fuequency;
        string airl_name;
        struct node* left;
        struct node* right;
      }
```

The slot order will similarly put in AVL tree

```
struct node{
        string airline;
        struct node* left;
        struct node* right;
      }
```

This gives us the best time complexity for both searching an element and adding an element. The inorder traversal of the slot AVL tree gives us the slot order.