SIES Graduate School of Technology

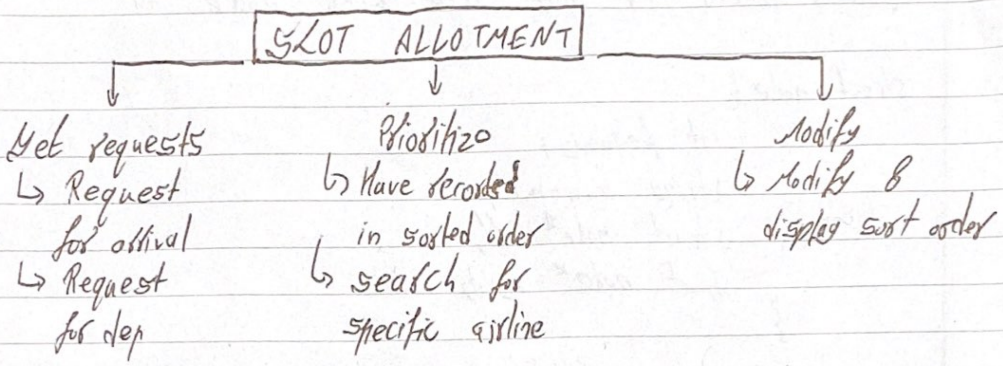
Subject : DS

Roll No.: 119A1076

Assignment No. 2

Date : 1/12/20

I Problem analysis & specification


SLOT ALLOTMENT
Get requests
↳ Request for arrival
↳ Request for dep
Prioritize
↳ Have records in sorted order
↳ search for specific airline
Modify
↳ Modify & display sort order

II Abstract Data types

Data: Records - Text string a_name
Int freq
Operation: sort() - To sort according to frequency
compare() - To search for particular airline.
Data: Request - Text string airline
Text string req (input : a/d)
Operations:- modify() - To append the slot arrangement
display() - To display ~~upd~~ 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 prioritizing. For this process, searching for an element in a data structure should have low

To fulfill the above requirements we have records stored in AVL tree with node as

```
struct node {
        int frequency;
        string a_name;
        struct node* left;
        struct node* right;
    }
```

slot order

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

This [illegible] gives us the best time complexity for both searching and adding.