**Prepare a zip file which includes only your solution files.**

·        You can use either your own or our code from the lab for data structure implementations.

**\*       You must use appropriate DATA Structures to solve the problems. Solving them with an array results in very poor marks.**

**Q1) Airline flight search (5pt).**Design and implement an airline flight search system.

A txt file stores each flight information in a single line. Format of the file is given below (and in flights.txt).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Date | Time | From | To | Carrier | Price |
| 06-05-06 | 10.35 | Ankara | Istanbul | THY | 235TL |
| 12-01-06 | 15.35 | Istanbul | Adana | Pegasus | 101TL |
|  |  |  |  |  |  |

Design and implement your own software. The requirements of the software are as follows. It must contain a user interface (command line is sufficient). Your software must allow user to

1)     Insert and remove flight information;

2)     Search for flights by **date,**however if the given date does not exist in the table show the **closest** before and after dates which have flight.

3)     Search for flights by **from city**;

4)     Search for flights with both **from city** and **date**.

5)     Search for flights between two dates.

6)     Search for flights less than a **given price** in a given date.

Considerations

·        Your searches must be efficient as possible in terms of order of complexity. O(log N) is usually possible.

·        You can generate the initial flight information file with random dates, cities, etc.

·        You can use any auxiliary java class Strings, GregorianCalendar, or design your own.

·        You can use multiple search data structures.