**Assignment 1: Flight scanner**

**Overview**

Create a program that fetches flights information from the web and analyses it, by using APIs from <https://openskynetwork.github.io/opensky-api/rest.html>.

**Part 1 - Bash script – Create DB**

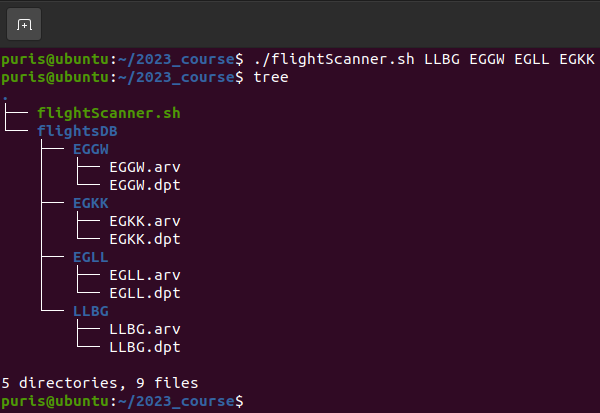
* Write a bash program that retrieves all arrivals and departures of the last 24 hours by given airports.
* The script shall retrieve the data in a json format.
* The script shall receive as input parameters, names of the airports in their ICOA code names.

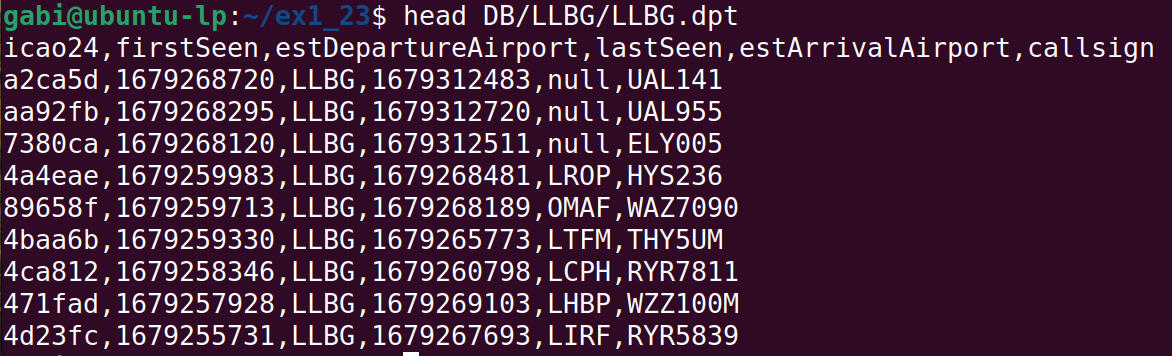
For example:

$ ./flightScanner.sh LLBG EGGW EGLL EGKK

* The script should output the following:
  + For each airport a folder named <airport\_name> containing two files in the following format <airport\_name>.dpt and <airport\_name>.arv .
  + <airport\_name>.dpt and <airport\_name>.arv shall contain all airport departing and arriving flight information, respectivly, in a **CSV format** with the first line containing the column info.

**Example:**





**NOTE 1:** Do not use jq utility.

**NOTE 2:** Handle errors.

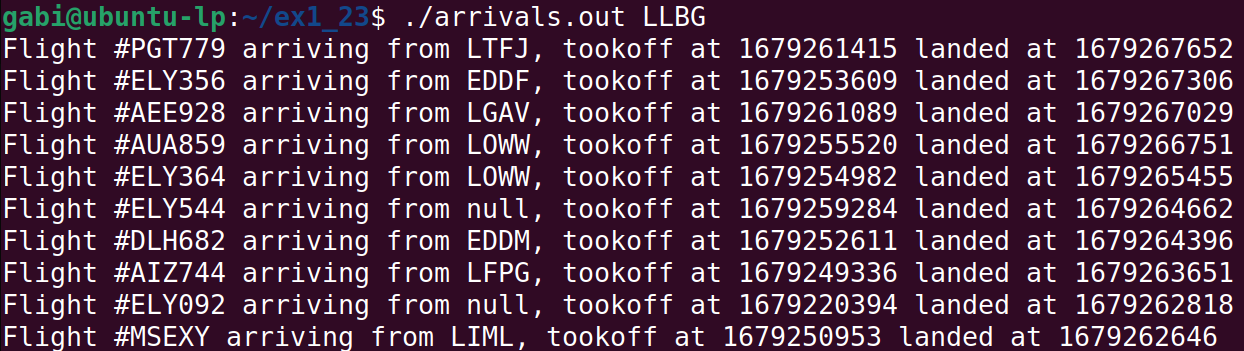
**NOTE 3:** Save columns needed for the next steps **only**.

**Part 2 - C code programs - Process data**

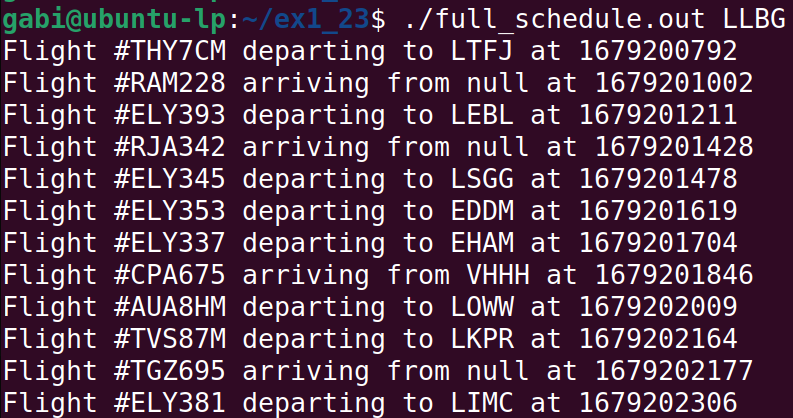
Write 4 C programs that read the data and perform the following:

* Write a utility library that provides functionalities like load\_db, get\_flights\_by\_airport\_name, get\_flights\_by\_callsign, rerun the script (to regenerate the DB), etc... and compiled as shared library (.so). All programs shall use it.
  + input handling and printing should **not** be done in the library
* Write 4 C programs(executables):

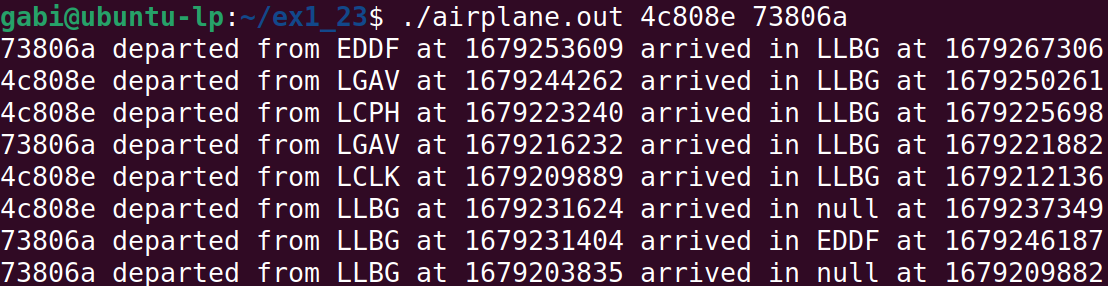
1. Receive airport ICOA code name**s** and output(print) a table of which airports have flights **to** the given airports and the flight details.



1. Receive airport name and output(print) the full airport schedule containing both arrivals and departures **ordered by time**.



1. Receive list of aircraft names(icao24) and output(print) all flights (departures and arrivals) that it did.



1. Program that updates the existing airports in the DB with recent data(rerun BASH script).

\* Flight number is represented by callsign

\* Bonus: print epoch time as readable time

**Part 3 - Makefile**

* Write a makefile that compiles the following 4 C programs (4 binaries and the shared library)

**How to submit?**

Zip all files together (only source files(.c, .h, .sh, Makfile), no binaries/output files(.o, .out, .so, etc...)).

We will unzip your zip file and then:

1) run the bash script and inspect the files it retrieved

2) Run “make” and make sure there are 4 .out programs, and 1 .so

3) Run each of the programs and expect the output described above

4) Run make clean and make sure all and only compilation outputs are deleted

\* Add a readme file with the names and id numbers of the group participants.