Summer 2017

MIS 6V99 – Special Topics – Programming for Data Science

Programming Assignment #1

Scheduling Analytics – Programmatically Creating and Optimizing an Airline Schedule

Kevin R. Crook

Scenario

A new startup airline is needing help with creating and optimizing a flight schedule. They have hired you as a data scientist to create and optimize a flight schedule. The airlines will be all business class and cater to business travel. All aircraft are configured exactly the same and can fly any route in the system interchangeably. The airline will serve Dallas Love Field (DAL), Austin Bergstrom (AUS), and Houston Hobby (HOU).

Aircraft and Tail Numbers

We have leased 6 aircraft. For sake of simplicity, we will assume all aircraft are configured exactly the same and can fly any route in the system and we will assume the “tail numbers” are as follows:

Aircraft “Tail Numbers”
T1
T2
T3
T4
T5
T6

Military Time and Minutes Since Midnight Calculations

All of the airports are on the same time zone. We will use a 4 digit military time format to represent times, with examples as shown below. Hint: for calculations involving time, it will be helpful to use an epoch of midnight and calculate the minutes since midnight = (hour * 60) + minutes, but the flight schedule should be printed in military time. To convert minutes since midnight to military time, hour = minutes since midnight div 60, minutes = minutes since midnight mod 60.

Civilian Time	Military Time	Minutes Since Midnight
6:00 am	0600	(6 * 60) + 0 = 360
7:21 am	0721	(7 * 60) + 21 = 441
11:59 am	1159	(11 * 60) + 59 = 719
12:00 noon	1200	(12 * 60) + 0 = 720
1:28 pm	1338	(13 * 60) + 38 = 818
2:24 pm	1424	(14 * 60) + 24 = 864
10:00 pm	2200	(22 * 60) + 0 = 1320

Noise Restrictions on First Departure and Last Arrival Times

Due to noise restrictions:

- flights cannot have a departure time of 0559 or earlier
- flights can have a departure time of exactly 0600
- flights can have an arrival time of exactly 2200
- flights cannot have an arrival time of 2201 or later

Flight Times (must be exact)

Flight Times are as follows (assume same flight time either direction, presented in "half alpha" order). Flights must be scheduled for exactly their flight time (no more, no less).

Airport	Airport	Flight Time in Minutes
AUS	DAL	50
AUS	HOU	45
DAL	HOU	65

Calculating Arrival Times

To calculate an arrival time for the schedule, use the following formula:

arrival time (minutes since midnight) = departure time (minutes since midnight) + flight time (minutes)

Example:

T1,DAL,AUS,0721,0811
departure time = 0721 = (7 * 60) + 21 = 441 minutes since midnight
arrival time = 441 minutes since midnight + 50 minutes = 491 minutes since midnight
491 div 60 = 8
491 mod 60 = 11
arrival time = 0811 military time

Number of Gates and Minimum Ground Time at Airports

We have secured gates at all airports. Each airport has a minimum ground time as follows. These are minimum times. Aircraft may be on the ground longer if designed.

Airport	Number of Gates	Minimum Ground Time in Minutes
AUS	1	25
DAL	2	30
HOU	3	35

Calculating Minimum Departure Times
(respecting the minimum ground times)

To calculate minimum departure time, use the following formula:

minimum departure time = arrival time (minutes since midnight) + minimum ground time (minutes)

example:

T1,DAL,AUS,0721,0811
T1,AUS,HOU,0836,0921
Arrival time = 0811 = (8 * 60) + 11 = 491 minutes since midnight
minimum departure time = 491 + 25 minutes = 516 minutes since midnight
516 div 60 = 8
516 mod 60 = 36
minimum departure time = 0836

Aircraft Repositioning for the Next Day

The schedule must start and end with the number of aircraft at an airport equal to the number of gates. It does not matter which specific tail number as all aircraft are configured the same, interchangeable, and may fly any route.

Restrictions on the Number of Aircraft on the Ground at an Airport at the Same Time

No airport may ever have more aircraft on the ground than the number of gates. An aircraft is considered on the ground from the arrival time (inclusive) until departure time (inclusive).

example:

T1,DAL,AUS,0721,0811
T1,AUS,HOU,0836,0921
In this example, T1 is on the ground in AUS from 0811 (inclusive) until 0836 (inclusive)
Since AUS has 1 gate, no aircraft can land with an arrival time during this period
A prior flight with a departure time of 0810 is permitted
A prior flight with a departure time of 0811 is not permitted
Another flight with an arrival time of 0836 is not permitted
Another flight with an arrival time of 0837 is permitted

Optimization Goals

Our optimization goals are to maximize the number of flights, utilize aircraft as evenly as possible, and utilize gates at airports as evenly as possible, and distribute flights among all 6 markets.

Python Program

You will write a single file Python program, create_flight_schedule.py, to accomplish them. The input data from the charts above may be hard coded into Python data structures. No input files will be required. The program must run successfully and create an output file flight_schedule.csv.

GitHub Repository ("repo")

Students must create a GitHub repository called mis_6v99_2017_summer as a private repository and grant access to the instructor account kevin-crook-ucb. They must create a directory in the repository called "assignment_01" and place the create_flight_schedule.py file in that directory. Students must check in code on a periodic basis and show periodic progress.

Format of the flight_schedule.csv output file

The flight_schedule.csv file should be created in the local directory. Do not use any directory path, as the auto grader will be looking in the local directory.

Below is an example of a snipped of a flight_schedule.csv file. Note there are no spaces and no enclosure quotation marks. Times should be printed in military time and always 4 digits. It should be sorted in the following order: tail_number, departure_time.

```
tail_number,origin,destination,departure_time,arrival_time
T1,DAL,AUS,0600,0650
T1,AUS,HOU,0715,0800
T2,DAL,HOU,0600,0705
T2,HOU,DAL,0740,0845
```

Grading Rubrics

Basic Criteria	Points
GitHub private repository was created correctly, instructor's account was given permissions, directory & files created correctly with exact names given, program runs without abend creating the specified output file, only the output file given by the instructor's run can be consider for further points	5
flight_schedule.csv in the correct format and sorted correctly (tail_number, departure_time)	5
All flights arrival and departure times are within noise restriction limitations specified	5
All flight times are exactly as specified and calculated according to the formula given	5
All ground times meet the minimum as specified and calculated according to the formula given (ground times can be greater than the minimum)	5
There is never more aircraft on the ground at an airport than the number of gates at that airport (see example above)	5
Aircraft repositioning for the next day – at the end of the day, each airport must have on the ground the same number of aircraft matching the number of gates – all aircraft are interchangeable so tail numbers do not matter	5

If and only if all of the Basic Criteria are met	Points
All aircraft fly at least 5 flights per day	10
All possible markets (AUS-DAL, AUS-HOU, DAL-AUS, DAL-HOU, HOU-AUS, HOU-AUS) have at least 3 flights per day	10

If and only if all of the Basic Criteria are met Additional Points Awarded Based on the Total Number of Flights in the Schedule	Points
Top 10%	10
Next 10%	9
Next 10%	8
Next 10%	7
Next 10%	6
Next 10%	5
Next 10%	4
Next 10%	3
Next 10%	2
Next 10%	1

Timing of Submission for Rank Grading

Since this is the first assignment, the submission time will not be considered in the tie breaker for rank grading. You may take the full time up until the due date and time.

Late Penalty

10% per day or fraction of a day. 1 second counts as a fraction of a day.

Technical Difficulties with Submission

If you encounter any technical difficulties with submission, in order to preserve your submission date and time, you must do all of the following:

- Have a sufficient history of work in GitHub.
- Immediately (within 1 minute) take a screen print which clearly shows the error you received on submission and also shows the date and time from your desktop clock.
- Email your instructor within 5 minutes of the error with the screen print attached.
- No further changes should be made to your GitHub after this email. All timestamps in GitHub must be before the deadline.
- If you are missing any of the above, please do not ask me for an exemption based on technical difficulties.

Resubmission Requests / submission errors

The assignment may be resubmitted any time prior to the due date without penalty.

Resubmissions after the due date will incur late penalties plus an additional penalty of 25%.

If student claim error in submission, they must have a sufficient work history in GitHub showing all work was completed prior to the dealine.