CPE 365 Lab 1-2: Why Databases?

Alex Ye, Nick Russo, Jon Kuzmich

I. Initial Decisions

The implementation language we chose was Python 2.6 because of it's simple data scanning and parsing functionality. Due to the small dataset, we decided we did not need any special environments or tools to get this done aside from the command line.

II. Selected Internal Architecture

The way we chose to manage data was a simple Python `list()`. With a simple dataset, we decided to have each index of the list to contain an array holding dynamic data representing a student. When we do a search, the list is then easily scannable through simple loops through the array.

III. Modifications to Part I

We elected to not start from scratch and to just add support for the new data file. We did this by simply making another Python dictionary that held teacher data. Then, when we were scanning in student information, we matched student to teacher using the room number as a key.

IV. Task Log

- Task: User Input Parsing.
    - Students performing it:
        - Alex
            - Start Time: 6:45 PM 9/24/16
            - End Time: 7:15 PM 9/24/16
            - Total Time Spent: 30 minutes
        - Jon
            - Start Time: ~7:00 PM 9/25/16
            - End Time: ~7:30 PM 9/25/16
            - Total Time Spent: ~ 30 minutes
        - Nick
            - Start Time: ~8:00 PM 9/25/16
            - End Time: ~8:30 PM 9/25/16
            - Total Time Spent: 30 minutes
    - Total Time Spent: 1hour 30 minutes
- Task: R4
    - Students performing it:
        - Nick

                - Start Time: ~7:30 PM 9/25/16
                - End Time: ~8:00 PM 9/25/16
                - Total Hours: ~30 Minutes.
        - Total Time Spent: 30 minutes
- Task: R5
    - Students performing it:
        - Alex
            - Start Time: 4:30 PM 9/24/16
            - End Time: 4:45 PM 9/24/16
            - Total Time Spent: 15 minutes
    - Total Time Spent: 15 mins
- Task: R6
    - Students performing it:
        - Jon
            - Start Time: ~7:00 PM 9/24/16
            - End Time: ~7:30 PM 9/24/16
            - Total Time Spent: ~30 Minutes
    - Total Time Spent: ~30 Minutes
- Task: R7
    - Students performing it:
        - Jon
            - Start Time: ~8:00 PM 9/25/16
            - End Time: ~8:30 PM 9/25/16
            - Total Time Spent: ~30 Minutes
    - Total Time Spent: ~30 Minutes
- Task: R8
    - Students performing it:
        - Nick
            - Start Time: ~7:00 PM 9/25/16
            - End Time: ~7:30 PM 9/25/16
            - Total Time Spent: ~1 hour.
    - Total Time Spent: ~ 1 hour
- Task: R9
    - Students performing it:
        - Jon
            - Start Time: ~7:30 PM 9/24/16
            - End Time: ~8:15 PM 9/24/16
            - Total Time Spent: ~45 Minutes
    - Total Time Spent: ~45 Minutes
- Task: R10
    - Students performing it:
        - Alex
            - Start Time: 4:45 PM 9/24/16

                - End Time: 5:00 PM 9/24/16
                - Total Time Spent: 15 Minutes
        - Total Time Spent: 15 mins
- Task: R11
    - Students performing it:
        - Alex
            - Start Time: 5:00 PM 9/24/16
            - End Time: 5:15 PM 9/24/16
            - Total Time Spent: 15 Minutes
        - Jon
            - Start Time: ~5:00 PM 9/26/16
            - End Time: ~5:15 PM 9/26/16
            - Total Time Spent: ~15 Minutes
        - Nick
            - Start Time: ~10:00 PM 9/26/16
            - End Time: ~10:15 PM 9/26/16
            - Total Time Spent: 15 Minutes
    - Total Time Spent: ~45 minutes
- Task: Reimplement data parsing for part 2:
    - Students performing it:
        - Alex:
            - Start Time: 5:00 PM 9/28/16
            - End Time: 5:30 PM 9/28/16
            - Total Time Spent: 30 Minutes
    - Total Time Spent: 30 Minutes
- Task: New search feature #1
    - Students performing it:
        - Alex
            - Start Time: 5:30 PM 9/28/16
            - End Time: 6:00 PM 9/28/16
            - Total Time Spent: 30 Minutes
    - Total Time Spend: 30 Minutes
- Task: New search feature #2
    - Students performing it:
        - Jon
            - Start Time: 8:00 AM 9/29/16
            - End Time: 8:20 AM 9/29/16
            - Total Time Spent: 20 Minutes
    - Total Time Spend: 20 Minutes
- Task: New search feature #3
    - Students performing it:
        - Nick
            - Start Time: ~10:00 PM 9/30/16

                - End Time: ~10:20 PM 9/30/16
                - Total Time Spent: ~20 Minutes
        - Total Time Spend:~20 Minutes
    - Task: New analytics feature #1
        - Students performing it:
            - Alex
                - Start Time: 5:30 PM 9/28/16
                - End Time: 6:00 PM 9/28/16
                - Total Time Spent: 30 Minutes
        - Total Time Spend: 30 Minutes
    - Task: New analytics feature #2
        - Students performing it:
            - Jon
                - Start Time: 10:15 AM 10/1/16
                - End Time: 10:50 AM 10/1/16
                - Total Time Spent: 35 Minutes
        - Total Time Spend: 35 Minutes
    - Task: New analytics feature #3
        - Students performing it:
            - Nick
                - Start Time: ~10:30 PM 9/30/16
                - End Time: ~11:00 PM 9/30/16
                - Total Time Spent: 30 Minutes
        - Total Time Spend: ~30 Minutes
    -

V. Testing

- Kinds of Testing
    - Unit Testing:
        - Student: Nick
        - Bugs found: 4
        - Total Time to Fix: ~15 mins
    - Manual Testing from Command Line:
        - Student: Nick, Alex, Jon
        - Bugs found: 15
        - Total Time to Fix: ~2 hour
- Total Time Testing/Debugging: ~2 Hours 15 Minutes

VI. Additional Comments

Alex

The purpose of this lab was for us to see what is required if we were to not use databases. Having worked with databases, writing functions to loop through data is much less efficient in both

runtime and time in implementing this code. These functions would have been doable in single line SQL queries in the fraction of the time if we have used a database.

Jon

For those not familiar working with databases like myself, this lab acts as a good introduction to working with data sets. The lab also revealed inefficiencies when dealing with data sets without a DBMS, such as a bloated codebase from similar code implemented in different search functions.

Nick

The purpose of this lab was to brush up on programming and skills and to work on problems, which would be easily solved with a DBMS, without a DBMS. Having experience with databases, this lab made me appreciate why DBMSs were created. Even for a simple library like Pandas, these task would be trivial and more efficient.

VII. Final Notes

This lab was a good introduction to the manual data processing we would need without the use of a DBMS. We were able to query data using basic, brute force methods to analyze the dataset. The timing functionality implemented shows us the time needed on such datasets and how it would not be a viable option for large and advanced datasets.