

Lab 1 Writeup

Justin Foxhoven & Matt Bryan

Initial Decisions

We chose to implement this lab assignment in Java, as that was the programming language we were both most familiar with. In addition to this, since Java runs in the JVM, it is system architecture agnostic and no special considerations need to be made when designing the program so that it will run successfully on lab machines. Java also has abundant high level support for a number of data structures and libraries that make file IO and parsing data relatively straightforward. All of these factors in conjunction made Java the best language for us when working on this lab.

Architecture

We used a list of lists as our primary internal data structure for a number of reasons. Originally, we considered using a Map as our primary data structure. If queries were only made using a primary key, this would work fine. However, given that we must query with a number of search parameters, a list of lists became a better option. With a list of lists, we will be able to iterate over each entry and check a particular index in each list for the piece of data we are querying for. This allows us to query with linear efficiency for any information associated with a given entry.

Task Log

Name	Student(s)	Start Time	End Time	Total Hours
Writeup	both	8 PM	9 PM	1 Hr
Design and Planning	both	11 AM	12 PM	1 Hr
File Parsing	both	11 AM	12 PM	1 Hr
Parsing User Input	both	11 AM	12 PM	1 Hr
Implementing Searches	both	2 PM	4 PM	2 Hrs
Implementing Commands	both	11 AM 2 PM 2 PM	12 PM 4 PM 4 PM	5 Hrs
Writing Tests and Testing	both	6 PM	8 PM	2 Hrs

Testing

Some testing was done during the implementation and both students were a part of finding the majority of these types of bugs as we were pair programming. This process lasted throughout the entirety of development and bugs were caught as commands were implemented. Most bugs we were able to fix right away, but a few bugs took upwards of half an hour to fix.

One major bug we had to fix after implementation was malformed output statements. Data was being emitted in a manner that was not consistent with the lab specs. While this wasn't unexpected behavior, it was still incorrect behavior and had to be fixed. In addition to this, a bug for computing the average GPA had to be fixed. In the case of a nonexistent grade level, the function would divide the total GPA, which was 0, by the total number of students in the grade, also 0. This was resulting in an error due to the division by 0.

Part 2 Modifications

Data Input Modifications

One of the main changes we made to Part 1 was changing our file parsing method into two separate methods for both input files. We chose to create two separate methods for ease of testing, and we feel it improved code readability as well.

This change was accompanied by the addition of a new method that returned a list of teachers based on a specific classroom. This new method was required to keep the functionality of many commands from Part 1. Since student and teacher data is now separated into two files, this method allowed certain methods to access teacher data that corresponded with specific students and classrooms.

Syntax Modifications

The Classroom command has two new flags: Student and Teacher. If the Student flag is specified, a list of students that use the classroom is printed. If the Teacher flag is specified, a list of teachers that teach in a given room is printed.

Example:

```
C[classroom]: <number> S[tudent]
```

```
C[classroom]: <number> T[eacher]
```

The Grade command has two new flags: Teacher and Analytics. If the Teacher flag is specified, all teachers that teach the specified grade will be printed. If the Analytics flag is specified, the GPAs of all students in that grade are printed.

Example:

G[rade]: <number> T[eacher]

G[rade]: <number> A[nalytics]

The Bus command has a new flag: Analytics. If the Analytics flag is specified, the GPAs of all students that use the specified bus route are printed.

Example:

B[us]: <number> A[nalytics]

The Teacher command has a new flag: Analytics. If the Analytics flag is specified, the GPAs of all students that have the specified teacher are returned.

Example:

T[eacher]: <lastname> A[nalytics]

There is also a single new command: Enrollments. This command outputs the enrollment of each classroom ordered by classroom number.

Example:

E[nrollments]

Final Notes

Our implementation of this lab meets all requirements as outlined in the lab specification to our knowledge. All errors should be handled gracefully and there should be no unexpected behavior.