CSCE 221 — Programming Assignment 5

Due: April 13th, 2017 at 11:59 pm

Program Description & Purpose This program contains the implementation of the Regex Library (Regular Expression) and a Hash Table in order to parse, store, and retrieve data from an input .csv file so that specified unique student UIN numbers and their corresponding grade can be retrieved in order to output the updated class roster with the respective grades appended to their corresponding UIN number. There are essentially 5 files, a main.cpp containing the driver program, the hash.h and hash.cpp files containing the hash table data structure, and of course, the input.csv and roster.csv files containing the student grades and the class roster. The program can be compiled and executed (directions shown down below in the **"How to Compile and Run"** section) in order to execute the process of updating the class roster with correct corresponding student grades, generating the "output.csv file".

Data Structures & Algorithm Description The primary data structures utilized in this program was the array data structure (utilized by the STL's string class & the Regex Library) in order to store strings for Regex parsing, and the Hash Table data structure, which is essentially a vector of linkedlists containing nodes that store a key-value pair (UIN, grade). Using the student UIN number as a hash key, the corresponding student grade can be retrieved in constant time. The Regex library used in the program also utilizes the array data structure in order to store the groups (or in the program's case) "matches", which are essentially just substrings acquired from the original parsed string and stored in an array for later access.

I/O description The input.csv file (which contains a list of student grades) was passed in through the I/O stream and was parsed line-by-line using Regex in order to retrieve the student UIN and student grade from each corresponding student which was then stored for later access in the Hash Table. Then, after all the data from the input.csv is stored in the hash table, the roster.csv was parsed line-by-line in order to retrieve the student UIN number as a hash key in order to search if their was a corresponding grade for that student UIN number located in the hash table. The program then generates the updated class roster with the appended student grades in the file **"output.csv".** There are a few assumptions made by the program, namely, it is assumed that each student UIN number is unique, and that the input is a .csv plain-text file so that the text can be passed into the stream and parsed with Regex. Testing was done and found that the corresponding grades from the input file had been appended to the correct student UIN numbers in the output.csv file generated by the program. (Shown Below, see the individual full .csv spreadsheet files in the program directory, from left to right, input.csv, roster.csv and output.csv)

C++/STL Features The primary C++ native features used in the program were the STL string class, the Regex library, and the I/O stream in order to store, parse, and access various data regarding input file, parsing said input data, and read/writing to files via the I/O streams.

Hash Table Statistics

- min - 1
- max - 1
- average - 1

Since the hash table is a vector of linkedlists that contain a node that encapsulates a key-value pair (UIN, grade), the computational results about the hashing are consistent with the expected constant runtime for the hashing algorithm because the search and insert functions in a list both run in constant time since there is only one element in each list.

How to Compile and Run

1. In a Unix terminal, navigate to the respective source directory **PA5/** using the **cd** command.
2. Once in the **PA5/** directory type **make all** to compile the respective program.
3. Run the executable for the Regex program by typing **./main** for the Regex program.
4. Compiled on build.tamu.edu host

Conclusion A greater understanding of hash tables, I/O streams, and Regex was acheived by implementing the data structure along with Regex in this program that updates a student grade roster by using a hash table and Regex/IO streams for parsing and reading/writing data.