**CS1342 – Spring 2023**

**Program 3 – Streams and Vectors**

In this program, you will utilize Streams and Vectors to create a which aggregates data about books! Your program must be submitted to GitHub by 6:00 AM CST on **Monday, March 6, 2023.** You will receive a 30 point penalty for any program submitted within 48 hours after the due date. Any programs submitted after 6:00 AM CST on **Wednesday, March 8, 2023** will receive a 0.

**Overview**

Data is often stored in CSV form (comma-separated-values). An example is shown below:

File - *Authors.csv*

ID,NAME,YEAR OF BIRTH,COUNTRY

1,J.K. Rowling,1965,United Kingdom

2,J. R. R. Tolkein,1892,South Africa

3,Stephen King,1947,United States

4,Dan Brown, 1964, United States

5,George R. R. Martin,1948,United States

Note that there is a row for the headers (ID, NAME, YEAR OF BIRTH, COUNTRY) and all values underneath are separated by commas. Each Author is separated by a new line.

As is common with data, an ID is used to provide a unique identifier for each author. That way if there just so happens to be 2 authors with the same name (like John Smith) we have a way of telling them apart.

Here is another example of a CSV file.

File – *Books.csv*

ID,AUTHOR\_ID,TITLE,GENRE,PUBLISHER

1,1,Harry Potter and the Sorcerer’s Stone,Fiction,Bloomsbury Publishing

2,1,Harry Potter and the Chamber of Secrets,Fiction,Bloomsbury Publishing

3,1,Harry Potter and the Prisoner of Azkaban,Fiction,Bloomsbury Publishing

4,2,The Hobbit,Fantasy,George Allen & Unwin

5,2,The Lord of the Rings: The Fellowship of the Ring,Fantasy,George Allen & Unwin

6,3,The Shining,Horror,Simon & Schuster

7,4,The DaVinci Code,Fiction,Simon & Schuster

This file has a dataset which lists out books written by a particular author. We can *link* or *join* this dataset with the first file (Authors.csv) by mapping the AUTHOR\_ID listed in Books.csv with the correct unique ID of the Authors table. In this way we perform what is known as an “Inner Join”

The purpose of an Inner Join is to return all records and columns from both tables by matching on a particular column, in this case Books.AUTHOR\_ID and Authors.ID.

The output would look something like this:

File – *Results.csv*

TITLE,GENRE,PUBLISHER,AUTHOR NAME,YEAR OF BIRTH,COUNTRY

Harry Potter and the Sorcerer’s Stone,Fiction,Bloomsbury Publishing,J.K. Rowling,1965,United Kingdom

Harry Potter and the Chamber of Secrets,Fiction,Bloomsbury Publishing, J.K. Rowling,1965,United Kingdom

Harry Potter and the Prisoner of Azkaban,Ficton,Bloomsbury Publishing, J.K. Rowling,1965,United Kingdom

The Hobbit,Fantasy,George Allen & Unwin,J. R. R. Tolkein,1892,South Africa

The Lord of the Rings: The Fellowship of the Ring,George Allen & Unwin, J. R. R. Tolkein,1892,South Africa

The Shining,Horror,Simon & Schuster,Stephen King,1947,United States

The DaVinci Code,Fiction,Simon & Schuster,Dan Brown, 1964, United States

…etc

Your Program will be implanting this Inner Join! Given 2 files (Authors.csv and Books.csv) your program will create a “join” of those 2 tables then output them to a file called Results.csv. You will be implementing the function declarations that we have provided for you (in addition to any others you find necessary) in order to create the result output file.

**Implementation Details**

Please read the implementation details carefully to ensure you receive full credit

* You may assume that the rows in the “Authors” table are unique
* You may NOT assume that the number of columns will be consistent. Your program should be able to correctly parse and output given input files or any size but consistent format
* You must implement and use the prototype functions that are given to you. Feel free to add any additional functions needed!
* You may assume no fields will be blank
* You must use a 2D vector to represent your data
* You must close your file stream properly when done using it
* Your output file must be called **Results.csv**

**Grading Rubric**

Correct Usage of File I/O – **10 points**

Files are appropriately closed - **5 points**

2D vectors are properly utilized **– 10 points**

All functions are implemented with function prototypes **– 5 points**

The readTable() function is implemented – **15 points**

The writeTable() function is implemented – **15 points**

The innerJoin() function is implemented – **20 points**

Any rows in the Authors table that do not exist in the Books table will not be included in the data set **- 10 points**

Code is formatted properly and has appropriate comments – **10 points**