Fall 2023 – CS 201 Data Structures and Algorithms
Homework-0

Maximum points: 100. Individual Work Only.

Due Date: September 14, 2023 before 11:59pm (**late submissions will get zero points**)

Objective

- Review programming in C++ and use C++ standard library functions

Problem Description

You are given an input file that contains movie names with year and actors separated by the character “/”. Write a C++ program that performs the following steps:

1. Read the name of the input file as a command-line argument (for example, ./a.out dbfile1.txt).
2. Open the specified file, create an object for each line in the file, add it to an appropriate collection of objects, and close the input file. Makes sure to implement a separate class that contains the name of the movie, year it was releases, and the actors along with appropriate methods required. Choose appropriate data structures to represent the movie name, year, and the actors. You are free to use any of the data structures provided by the C++ standard library (i.e., std namespace).
3. Using the sort method provided by the C++ standard library (std::sort) sort the collection of objects based on the movie name. If the movie name happens to be the same then use the year released to break the tie.
4. Write the sorted collection based on the movie names to a file called “dbfile1ByName.txt” in the same format as the input file (movie name (year)/actor-1/actor-2/…/actor-n) if the input file name is dbfile1.txt.
5. Using the sort method provided by the C++ standard library (std::sort) sort the collection of objects based on the year the movie was released. If there are many movies released in a given year then use the name of the movie to break the tie.
6. Write the sorted collection based on the year the movie was released to a file called “dbfile1ByYear.txt” in the same format as the input file (movie name (year)/actor-1/actor-2/…/actor-n) if the input file name is dbfile1.txt
7. After testing you program for correctness using the input file dbfile0.txt, use the different input files provided in Blackboard to complete the table below, include the table in the report, and an analysis of the execution times for different input files.

Input File	# of records	Time taken to create the collection	Time taken to sort based on the movie name	Time taken to sort based on the year	Total time taken
dbfile1.txt	442	0.004867 sec	0.000328 sec	0.00029 sec	0.005485 sec
dbfile2.txt	7065	0.075358 sec	0.006816 sec	0.006508 sec	0.088682 sec
dbfile3.txt	14129	0.149276 sec	0.01823 sec	0.020863 sec	0.188369 sec

You can use appropriate C++ standard library functions to measure the time taken (for example, std::chrono).

8. Include a short description in the report that describes the rationale for your choice of the data structures for storing the movie name, year, and the actors and also the choice of the data structure used to create the collection of objects.

Program Documentation and Testing

1. Use appropriate variables names and indentation in your source code.
2. Include meaningful comments to indicate various operations performed by the program.
3. Programs must include the following header information within comments:

```
/*
  Name:
  Email:
  Course Section: Fall 2023 CS 201
  Homework #:
  Instructions to Compile:
  Instructions to Execute:
*/
```

Submission

Upload only the source files (.h or .hpp or .cpp or .cc files) and the report (Word or PDF file) to Blackboard in the assignment submission section for this homework. Do not upload zip/tar files to Blackboard, upload individual source files (no object files or executables) and the Word or PDF file for the report.

Grading Rubrics

The following rubrics is used for grading:

Description	Points
1. Correct implementation of the solution following all the requirements (steps 1 – 6 in the problem description)	150
2. Completing the table and explanation of results (step 7)	30
3. Justification for the choice of the data structures used (step 8)	20