Frank Hulmes
Alvie Hossain

CISC220 Final Project: The Movie Recommender

Project Summary:

This project is an example of data structures implemented in a movie recommendation application. The application builds and organizes a database of IMDB rated movies using linked lists, arrays, binary heaps, and the heapsort sorting algorithm. This database is originally stored in a text file, and then read into the program. The application asks the user to input either a movie title, or a movie genre. Based on this input the program goes through the heapsort-ed data structures and either recommends the best movies in that genre, or movies similarly rated by IMDB depending on user preference. We simplified the project by using only 5 movie genres, and stuck to inputting movies released in 2016.

Data Structures Used:

Array/Binary Heap

When processing the data, we wanted to be able to quickly and reliably pick out movies that are similar to the movie selected or be the best movie in each genre. We thought the best way to accomplish both of these at the same time without recommending the same movies over and over would be to use a binary heap in the form of an array. This way, we can quickly pick out the movies with the highest rating (highest priority) and be able to choose movies that are close to the selected movie in the tree to recommend similar movies.

Heapsort

Since we decided to use a binary heap to process the data after first putting it into a linked list, the best way to sort this data based on rating would be to use a heapsort. This sorting algorithm is fairly efficient, with most scenarios taking O(nlog(n)) time, making it consistent regardless of the order of the movies entered.

Linked Lists

To efficiently access the data in this application, an ordered linked list seemed like the best way to access movie recommendations for users. The idea behind the application is that users when watching a movie, will like another movie that is similarly rated. So because our movies are organized in a sorted genre-based linked lists, we can easily recommend a specific movie node’s previous nodes, and next nodes. Also, using a linked list lets us easily add or remove movies at any time.