COM 212 Project

We have this really cool idea to create a database of movies that registered customers can watch. Call it NetworkFilms, or something like that. Before we add the actual content, we need to think about what features we want the service to have, and accordingly, how the movies and customers need to be able to interact.

Each customer record must include:
  - Their name
  - A credit card number
  - An email address where we can reach them
  - The customer's wish-list of what to watch next (up to twenty movies) with the following requirements:
      - Access the next movie in constant time (if it is marked as no longer available, a note should be displayed, the movie deleted, and the next movie shown).
      - After access, give an option to delete (constant runtime).
      - Add new movies to the back of the list (constant runtime).

Each movie archive must include:
  - A title
  - A release date (yyyymmdd ex: 20160421 = April 21st, 2016)
  - A unique ID code (at least 5 digits long)
  - The movie's Rotten Tomatoes popularity score (0-100, not subject to change)
  - A Boolean delineating whether or not we hold the movie

It is vital that this underlying foundation of the program works as fast as possible.

With this in mind, we have to be able to access each customer's information quickly when we want to change any of their information. Specifically, we need to access any customer record in an average less than $O(n)$ runtime. We can track them based on their credit card numbers.

The movies are a bit trickier – although we only keep a maximum of 255 available movies, the following access requirements are specified:
  - The administrators need to be able to see the least rated available movie (Rotten Tomatoes popularity score) in constant time and delete/add new movies in less than $O(n)$ time. They will periodically check the least rated movie and remove it when no longer desired to make sure there is always room for new movies coming in. This is the only way movies will be removed from availability from our system.
  - The customers need to be able to:
      - Access any movie that we have ever had and the Administrators want to keep (including available and not available) in order of release date in less than $O(n)$ runtime. Administrators should be able to delete movies from this list, although they will never delete movies marked as available.
      - Access any movie by ID in less than $O(n)$ runtime.
      - Print to screen all movies in order of release date.

Since we are still in the testing phase, actual customers won't be able to directly access the service. Instead, an admin from our company will process all requests (i.e. there is no log-in system for you to worry about. You can just assume that the person using the program has received each instruction from an actual customer, or has permission to alter the movie database).

Extra features that we'd like included in the future are:
  - A log-in system that has a separate menu for users and admins (+5)
  - A "have watched" list of movies each customer has seen (+5)