

# COMMUNITY LIBRARY APPLICATION

*Alex Myers  
Nathan Coyne  
Rona Rosal*

## Contents

Contents	1
Introduction	4
Algorithm Design	4
Empirical Algorithm Analysis	5
Step 1: Understand the experiments purpose	5
Step 2: Decide on efficiency metric to be measured	5
Step 3: Decide on the characteristics of input sample	5
Step 4: Prepare a program implementing the algorithm	5
Step 5: Generate sample of inputs	5
Step 6: Run the algorithm on sample inputs and record data observed	6
Step 7: Hypothesize data observed	8
Software test plan and test results	9
<b>Main Menu</b>	9
<b>Staff Menu</b>	10
1. Add new DVDs of a movie to the system (Option 1)	10
A new movie not in current system (Branch 1)	10
Successful Attempt:	10
Unsuccessful Attempts:	10
Whitespace or Empty Movie Name	10
Invalid Genre Input	11
Invalid Classification Input	11
Invalid Movie Length	12
Invalid Initial Copies	12
An existing movie in current system (Branch 2)	13
Successful Attempt:	13
Unsuccessful Attempts:	13
Unexpected New Movie	13
Invalid Number of Copies to Add	13
2. Remove DVDs of a movie from the system (Option 2)	14
Successful Attempts:	14
Unsuccessful Attempts:	14
Movie doesn't exist	14
Invalid Number of Copies to Remove	14
Movie has less available copies than can be removed	15
3. Register a new member with the system (Option 3)	15

Successful Attempt:	15
Unsuccessful Attempts:	15
Whitespace or Empty First Name	15
Whitespace or Empty Last Name	15
Invalid Phone Number	16
Invalid Pin Number	16
4. Remove a registered member from the system (Option 4)	16
Successful Attempt:	16
Unsuccessful Attempt:	16
Member doesn't exist	16
Member is currently borrowing movies	17
5. Display a Member's Contact Phone Number (Option 5)	17
Successful Attempt:	17
Unsuccessful Attempt:	17
Member doesn't exist	17
6. Display Borrowers (Option 6)	17
Successful Attempts:	17
Unsuccessful Attempt:	18
<b>Member Menu</b>	18
1. Display all information about all the movies DVDs (Option 1)	18
Successful Attempts:	18
Unsuccessful Attempts:	18
2. Display the information about a specified movie (Option 2)	19
Successful Attempt:	19
Unsuccessful Attempts:	19
Movie does not exist	19
3. Borrow a movie DVD (Option 3)	20
Successful Attempt:	20
Unsuccessful Attempts:	20
Movie does not exist	20
User already borrowing the movie	20
User is borrowing 5 movies already	20
There are 10 borrowers of a movie already	20
4. Return a movie DVD (Option 4)	20
Successful Attempt:	20
Unsuccessful Attempts:	21

Movie Doesn't Exist	21
User is not Borrowing the DVD specified	21
5. List currently borrowed DVDs by the current logged in member (Option 5)	21
Successful Attempts:	21
Unsuccessful Attempts:	21
6. Display top 3 movies that have been borrowed the most (Option 6)	21
Successful Attempts:	21
Unsuccessful Attempts:	22

## Introduction

The purpose of this project is to develop a Community Library System to manage its movie DVD's. The application aims to store, manage, and manipulate data through a console application. We store movies, movie information, customers and customer data. The efficiency of the top 3 rented movies is then looked at with an empirical analysis.

The application has a Main Menu page that allows for staff login, user login or exit.

First, a staff login option is available to all staff. A staff member is allowed to add or remove a movie or add, subtract copies of that movie. They also register or remove a member in the system as well as display a member's contact details and the members who are currently renting a movie.

Second, a member can login with first name, last name and pin, to get user menu. Members are allowed to display all movie information or all movies. In addition, members are also allowed to borrow and return a movie. They can see their borrowed movie as well as the top 3 most borrowed movies by the members in the library.

## Algorithm Design

For this algorithm, we transform the movie collections into an array and sort the top 3 borrowed movies by the number of borrowings. Since number of borrowings is an integer the pseudo code removes the need to represent a movie and just uses integers.

The basic operation appears in 3 places,  $A[i] > max$ ,  $A[i] > nd$ , and  $A[i] > rd$ . We then insert the counter before these basic operations execute. The algorithm can perform different numbers of basic operation at each iteration.

### **ALGORITHM** *Top3Movies(A[0..n - 1])*

```
// Finds the top 3 borrowed movies, by the integer no of borrowings.  
// Input: An array A[0..n - 1]  
// Output: Prints the top 3 movies by number of borrowings  
max ← 0  
nd ← 0  
rd ← 0  
for i ← 0 to n - 1 do  
    if A[i] > max  
        rd ← nd  
        nd ← max  
        max ← A[i]  
    else  
        if A[i] > nd  
            rd ← nd  
            nd ← A[i]  
        else  
            if A[i] > rd  
                rd ← A[i]  
Write(“ 1st : ” and max)  
Write(“ 2nd : ” and nd)  
Write(“ 3rd : ” and rd)
```

## Empirical Algorithm Analysis

### Step 1: Understand the experiments purpose

The goals of analysing the algorithms empirically are as follows:

- To compare the efficiency of the several algorithms for solving the same problem
- To compare different implementation of the same algorithm
- To develop a hypothesis about the algorithm's efficiency class.
- To ascertain the efficiency of the algorithm implementing the algorithm on a specific

### Step 2: Decide on efficiency metric to be measured

- The number of times the algorithms basic operation is executed is based on the size of the array.
- The execution time of the algorithm's implementation.

### Step 3: Decide on the characteristics of input sample

- The problem size parameter(s) is the size of the array or how many movies are in the collection turned into an array.
- The top 3 movies by movie borrowings could be replaced with a random array of integers as movie borrowings is just an integer. The basic operations are the 3 if statements so we put a count above them.

### Step 4: Prepare a program implementing the algorithm

**Programming Language used:** C#

*A count for each basic operation was placed in the algorithm and a timer, we took the average of the 20 iterations per array size.*

### Step 5: Generate sample of inputs

Sample inputs: 1000 -100000, going up in intervals of 1000.

## Step 6: Run the algorithm on sample inputs and record data observed

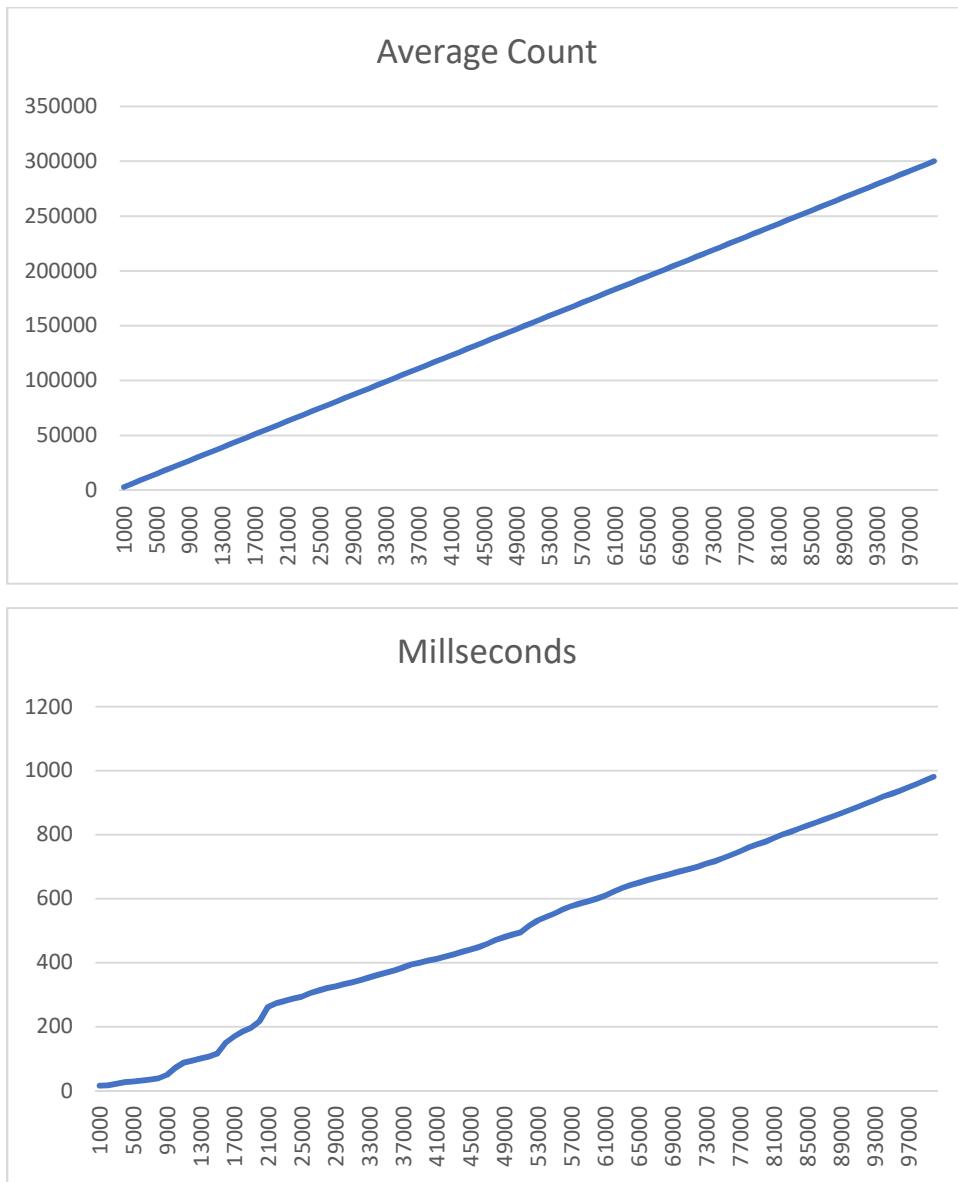
2146724743	2143127743	2142695560	Size = 6000; Average Count = 17968.05	2147362106	2147156615	2147042482	size = 8000; Average Count = 23967.25	2147359454	2147217176	2147146272
2147472537	2147272235	2144387317	2146821283	2145844173	2143326699	2147417349	2146823476	2146781652		
2147432073	2144637967	2143690178	2147034933	2146616208	2145513054	2146827292	2146679798	2145916276		
2147279748	2147004203	2146595625	2147283600	2146926545	2146516228	2147258012	2147069520	2146164438		
2146452097	2144567428	2144558011	2145398685	2145292596	2144898869	2147355454	2147241614	2146569672		
2147105424	2145868177	2144964546	2147268198	2146832667	2146071021	2146948730	2146850092	2146776776		
2147330662	2144146220	2143055692	2146708683	2146100660	2144542282	2147368330	2147214822	2146931856		
2146802115	2146187309	2144231534	2147322130	2147307132	2144919255	2146804071	2146684302	2145180735		
2146886408	2146847354	2146830672	2147427685	2146358374	2145807551	2147465583	2147451893	2144443070		
2147393933	2144914011	2142542532	2146866524	2146674685	2145692518	2147378458	2146750962	2145880041		
2146469195	2145338682	2143350651	2147411870	2147040873	2146987846	2147214032	2147848955	2146698746		
2147280703	2145323631	2145222050	2147107899	2146661343	2146025887	2147291230	2146974678	2146548450		
2146726870	2146709993	2142338115	2147021493	2146598743	2146395383	2147348400	2147247473	2146459125		
2146668203	2145414013	2144787609	2147305250	2147171634	214680312	2147151860	2146992454	2146368043		
2146433217	2143589496	2142542733	2147346558	2147028808	2146001151	2147391747	2147336035	2147035199		
2147385117	2147351708	2147325307	2147293727	2147212630	2146547697	2147423094	2147336653	2146266260		
2147221150	2147171577	2140209848	2146806000	2146428847	2145570633	2147921888	2147267952	2147182250		
2147088848	2143672176	2142177739	2147248723	2146746513	2145917548	214748889	2146921771	2144622638		
2145522959	2144118485	2143614209	2147101691	2147036675	2146292051	2147307297	2147130674	2145664710		
2146699789	2143982388	2139591498	2147416001	2147219606	2146717003	2147351538	2147210599	2146719495		
2147258577	2146798891	2143439656	Size = 3000; Average Count = 8969.6	2147460828	2147893260	2146577635	2147373350	2147328993	2146203463	
2146710181	214640139	2145548800	2147334054	2146946421	2146744443	2147381767	2147276256	2146225896		
2147483276	2147229853	2146412983	2147432878	2146102465	2145324516	2147436020	2147290901	2147242211		
2147401564	2146586206	2146387268	2147170769	2146282035	2146262422	2147399478	2147121404	2145856946		
2147438114	2147437253	2145924885	2147441961	2146398479	2145030972	2146795927	2146314046	2146259676		
2146796001	214960097	2143952842	2146775060	2145679533	2145425390	2147482444	2147157341	2145716924		
2147390932	2147179863	2147164042	2147406369	2147124170	2146544625	2147389572	2147303296	2147130214		
2146688439	2146419001	2145377523	2147406212	2146648819	2146485858	2147376833	2147077053	2145861182		
2146658731	2146433210	2143756278	2147173399	2146758411	2146597447	2147224418	2145738950			
2147199342	2146999195	2146374285	2147436178	2147092512	2146784970	2147310278	2147176905	2146601841		
2146752132	2145841323	2145373463	2147312606	2147100810	2146422462	2147302430	2146678855	2144981675		
2147241561	2147118278	2146581789	2147417828	2147338204	2146541454	2146839846	2146616210	2145296990		
2146802419	2146504944	2146259215	2147478769	2147389375	2147164625	2147458600	2145899105	2145559288		
2147388704	2146655908	2144694914	2147082620	2146981321	2146619187	2147407001	2146994254	2146720081		
2146650632	2144863694	2143828223	2147310477	2147126272	2146961972	2147122951	2146644403	2146430705		
21472766824	2146746221	2145840874	2147377235	2146661005	2146548392	2147436083	2147164285	2147009241		
2147085807	2145265622	2143903509	2147052039	2146743429	2146690661	2147401441	2147321427	2147174686		
2145975287	2143664448	2142556646	2147064974	2146513809	2146063284	2147409342	2145970534	2145293460		
2147428842	2145302432	2144354343	2147147284	2147281974	2147198587	2147366630	2146681252	2146303566		
2147088934	2146692800	2144072864	2147149674	2146905453	2146599018	2147176919	2147057591	2145778506		
Size = 10000; Average Count = 29965.35	2147410706	2147376875	2147039819	2147374351	2147316057	2147066605				
2147274040	2147146873	2146796888	2147481960	2147401235	2146612783	2147478319	2147202385	2146373806		
2147308181	2147085521	2146887006	2147372466	2146921020	2145648474	2147349671	2147013359	2146972666		
2147098941	2146956101	2146651803	2147458516	2147434860	2147136974	2146926388	2146650332	2146175398		
2147419674	2147301555	2146207684	2147334524	2146900880	2146555260	2147428248	2147063139	2146321208		
2147373188	2146958020	2146450295	2147398320	2147121635	2146989889	2147064514	2146994585	2146844401		
2147416496	2147088928	2146849414	2147239632	2147200734	2146646600	2147351284	2147235968	2147141365		
2147252777	2147106067	2145432157	2147428272	2146958885	2146675494	21474264947	2147150209	2146900508		
21472211976	2146594642	2145951280	2147355067	2147185848	2146785580	2147461514	2147122859	2147025640		
2147242295	2146485699	2146179367	2146976269	2146727289	2146452495	2147204578	2146965269	2145281730		
2147392622	2147322577	2146903139	2147475978	2147280276	2147254567	214719773	2146416902			
2147442033	2147324823	2147188818	2147310210	2147166823	2147127990	2147458339	2147310631	2146783687		
2147196904	2146768195	2146006326	2147470839	2147298466	2145704802	2147427717	2147147495	2146199975		
2147392492	2147246622	2147195557	2147394714	2146974788	2146057420	2147441798	2147354363	2147268253		
2147267088	2147176656	2146353014	2147479316	2147446806	2147175217	2147453571	2147169200	2146707080		
2147029712	2146963321	2146488896	2147246702	2146958971	2146678979	2147176778	2147058058	2146519737		
2146978973	2146819291	2147604988	2147460268	2147249565	2146829124	2147483198	2146621354	2146588029		
2147240814	2147148419	2146398594	2147102555	2147047651	2146901524	2147483029	2147007758	2146946648		
2147408889	2147374988	2145821958	2147480765	2147395924	2146762133	2147299262	2146886357	2146511772		
2147428453	2147172021	2147022050	2147376786	2147432416	2147256387	2147450000	2147311168	2147140776		
Size = 11000; Average Count = 32965.8	2147177515	2146893229	2147410876	2147374351	2147316057	2147066605				
21474120024	2146869908	2146149592	2147448726	2146975322	2146830379	2147421990	2147349628	2147256580		
2147442629	2146278576	2146266617	2147370771	2147364732	2147311447	2147310858	2147217806	2147166162		
2147355182	2146820853	2146294583	2147223506	2147061935	2146752422	2147446227	2147253194	2147014166		
2147259397	2146605952	2145787715	2147430349	21473743043	2147296633	2147446327	2147249924	2147008460		
21474747407	2147183585	2146902811	2147416382	2147212033	2147208865	2147311348	2147288840	2146868199		
2147373376	2146363051	2146060871	2147379992	2146475798	2145900483	2147417645	2147207653	2146604477		
2147453008	2147207032	2146995472	2147454733	2147409867	2147351771	2147323949	2147290381	2147145957		
2147377062	2147177726	2146996369	2147447084	2147233989	2146455196	2147280890	2147169122	2146777245		
2147181765	2146765291	2145968487	2147448434	2147306927	2147276974	2147459089	2147355704	2147248744		
2147411128	2147407262	2145395770	2147467675	2147187934	2147016122	2147118932	2146913759	2146355423		
2147462115	2147021555	2146598130	2147425472	2147225984	2145995131	2147259977	2147255951	2147252357		
2147457794	2147235810	2146493242	2147261866	2146854920	2146703344	2147472923	214718			

2147238876	2146969417	2146886611
2147435241	2147317809	2146939412
2147477387	2147344612	2146870116
2147413179	2147482475	2147378639
2147324878	2147263886	2147001233
2147266064	2147004062	2146865124
2147473526	2147294402	2146573658
2147470339	2147273335	2147188890
2147442247	2147275338	2147246297
2147364749	2147245908	2147197795
2147438677	2147417086	2146195268
2147240748	2147186545	2146093013
2147436212	2146967129	2146954370
2147232001	2147095887	2146991068
2147278187	2147227748	2146751313
2147477497	2147234596	2147179602
2147432243	2147384628	2147157391
2147346220	2147113605	2146776755
2147247727	2147221161	2146846978
2147402583	2147243249	2147187676
Size = 18000;	Average Count = 53962.7	
2147480159	2147180343	2147098030
2147323134	2147167313	2147119323
2147386369	2147269849	2147148461
2147460070	2146797784	2146562906
2147101154	2147036863	2146808414
2147417194	2147222042	2146671290
2147451965	2147230527	2146898600
2147251467	2147150540	2146782973
2147234552	2147180605	2146970207
2147399972	2147278802	2146640162
2147398050	2147390391	2147034895
2147203178	2147103773	2147052632
2147387861	2147329377	2147252989
2147311087	2147090677	2146982399
2147454976	2147284063	2147223425
2147198161	2146846843	2145928671
2147423116	2147062399	2146831440
2147220794	2146920719	2146889764
2147380612	2147188793	2147118879
2147431625	2147430916	2147163659
Size = 19000;	Average Count = 56962.6	

2147333205	2147193462	2147142252
2147469927	2147040029	2146776102
2147381207	2147331481	2147221275
2147418595	2147195543	2146068996
2147457101	2147384631	2147199335
2147347599	2147113387	2147092924
2147455391	2147207907	2146924325
2147467694	2147440783	2146931192
2147309667	2147187637	2147132497
2147470485	2146910480	2146743805
2147341894	2147054670	2146881923
2147378727	2147324883	2146880278
2147323051	2147050620	2146927749
2147435438	2146867816	2146865980
2147463964	2146898094	2146476259
2147385320	2147253264	2147010246
2147469004	2147001678	2146475878
2147452178	2147066949	2146492400
2147318895	2147091615	2146719090
2147482999	2147473650	2146932266
Size = 20000;	Average Count = 59963.1	

### Step 7: Hypothesize data observed

We hypothesize a likely efficiency class of an algorithms based on the following empirical observations of its basic operation's count:



We had arrays of different sizes full of random numbers to show the number of borrowings. The cases were size of array 1000 to 100000, going up in intervals of 1000, run 20 times. The average time and count were returned then graphed against the array size.

Taking samples from the test cases in the array, we will use the following to calculate the ratio and determine the likely efficiency class of the algorithm.

We analysed the data observed to form a hypothesis about the algorithm's average case efficiency.

	n	average count	average time			n	average count	average time
n	50000	299962.6	981		n	30000	89967.35	333
2n	100000	149966.95	488		3n	90000	269966.5	877
2n/n	2	2.000191375	2.010246		3n/n	3	3.000716	2.633634

To calculate  $g(n)$  we need to see the relationship between  $2n/n$  and  $\text{count}(2n)/\text{count}(n)$  and  $\text{time}(2n)/\text{time}(n)$ . We can see that the ratios for count and time are very close to 2 so there is a linear relationship between the size of the array and the time it takes or number of operations performed. Time is slightly less than 3 for  $3n$  but time mostly needs to be tested at higher levels so it can be a little off unless you use larger values. But we can see that the  $g(n)$  is  $n$  so now we will test for  $g(n)=n$ .

n	count(n)/n	time(n)/n
64000	2.999464844	0.01003125
65000	2.99948	0.01
66000	2.999458333	0.009969697

Since the ratios are all approximately equal to 3 and 0.01 it's a constant for each one so therefore it is likely  $O(n)$ .

## Software test plan and test results

All unsuccessful attempts will not terminate the adding process but rather re-ask the question again until successful most of the time. Each unsuccessful attempt assumes the previous inputs follow the previous steps successfully corresponding to the successful attempt above it.

## Main Menu

The main menu is the first thing which the user sees when opening the program and is fairly self-explanatory. Entering the relevant number option will make you go to the option corresponding to the number or exit if it is 0. The same logic applies to all 3 menus.

```
=====
Welcome to the Community Library Movie DVD Management System
=====

===== Main Menu =====

1. Staff Login
2. Member Login
0. Exit

Enter your choice ==> (1/2/0)
-
```

## Staff Menu

```
===== Staff Menu =====
1. Add new dvds of a new movie to a system
2. Remove dvds of a movie from the system
3. Register a new member within the system
4. Remove a registered member from the system
5. Display a members contact number given the members name
6. Display all members who are currently renting a particular movie
0. Return to main menu

Enter your choice ==> (1/2/3/4/5/6/0)
-
```

1. Add new DVDs of a movie to the system (Option 1)

### A new movie not in current system (Branch 1)

When a new movie title not in the system is entered, the system will ask for extra information to initialise the movie.

The movie **Titanic** was input and due it being a new movie the program then asks for the input of the genre, classification and initial copies.

#### Successful Attempt:

**Titanic** with the genre **1** (Drama), classification of **2** (PG), duration of **60** minutes and initial copies of **3** is added.

```
Please enter the title of the new movie to add
Titanic
Please choose the movie genre
1. Drama
2. Action
3. Western
4. History
5. Comedy

Enter your choice ==> (1/2/3/4/5)
1
Please choose an option corresponding to the classification of the movie
1. G
2. PG
3. M
4. M15+

Enter your choice ==> (1/2/3/4)
1
Please enter the number of minutes for the movie
60
Please enter the number of copies initially available for the movie
3
Titanic successfully added
```

#### Unsuccessful Attempts:

##### Whitespace or Empty Movie Name

Movie names cannot be empty or contain only spaces. In this example, the movie name was first entered as nothing, and then contained a number of spaces.

```
Please enter the title of the new movie to add
Movie must have a name
Please enter the title of the new movie to add
Movie must have a name
Please enter the title of the new movie to add
```

## Invalid Genre Input

The genre must be entered as a number from 1-5 and if it is not, it counts as unsuccessful. In the example below, the word **Drama** was entered instead of **1** so it re-printed the menu. **8** also failed as it wasn't in the range of 1 to 5.

```
Please enter the title of the new movie to add  
Titanic  
Please choose the movie genre  
1. Drama  
2. Action  
3. Western  
4. History  
5. Comedy  
  
Enter your choice ==> (1/2/3/4/5)  
Drama  
Please choose the movie genre  
1. Drama  
2. Action  
3. Western  
4. History  
5. Comedy  
  
Enter your choice ==> (1/2/3/4/5)  
8  
Please choose the movie genre  
1. Drama  
2. Action  
3. Western  
4. History  
5. Comedy  
  
Enter your choice ==> (1/2/3/4/5)
```

## Invalid Classification Input

Similarly to genre, the classification must be entered as a number from 1-4. In this example **PG** was entered instead of **1**, and **8** failed as well because it wasn't in the range of 1-4.

```
Please enter the title of the new movie to add  
Titanic  
Please choose the movie genre  
1. Drama  
2. Action  
3. Western  
4. History  
5. Comedy  
  
Enter your choice ==> (1/2/3/4/5)  
1  
  
Please choose an option corresponding to the classification of the movie  
1. G  
2. PG  
3. M  
4. M15+  
  
Enter your choice ==> (1/2/3/4)  
PG  
Please choose an option corresponding to the classification of the movie  
1. G  
2. PG  
3. M  
4. M15+  
Enter your choice ==> (1/2/3/4)  
8  
Please choose an option corresponding to the classification of the movie  
1. G  
2. PG  
3. M  
4. M15+  
Enter your choice ==> (1/2/3/4)
```

## Invalid Movie Length

The movie length must be in positive, whole minutes, must not include ‘minutes’ at the end and just be the raw number in numerical form. It also cannot be 0 minutes as a movie must run for at least some amount of time. In the example below **-4**, **grgrt**, and **4.5** were all entered. These are all seen as invalid as they are not follow the requirements.

```
Please enter the title of the new movie to add
Titanic
Please choose the movie genre
1. Drama
2. Action
3. Western
4. History
5. Comedy

Enter your choice ==> (1/2/3/4/5)
1
Please choose an option corresponding to the classification of the movie
1. G
2. PG
3. M
4. M15+

Enter your choice ==> (1/2/3/4)
1

Please enter the number of minutes for the movie
-4
The movie duration can't be negative minutes long
Please enter the number of minutes for the movie
grgrt
The movie duration must be in whole minutes
Please enter the number of minutes for the movie
4.5
The movie duration must be in whole minutes
Please enter the number of minutes for the movie
```

## Invalid Initial Copies

The number of copies initially available must be a positive, whole number. Similar to the movie length it must not include the ‘copies’ word at the end and just be the raw numerical number. Also, because of how the movie library operates, it cannot have 0 copies initially available because it is then not allowed in the system. In the example below **-3**, **0**, **Zero**, **Five**, and **4 copies** were all entered and all do not satify the conditions.

```
Please enter the title of the new movie to add
Titanic
Please choose the movie genre
1. Drama
2. Action
3. Western
4. History
5. Comedy

Enter your choice ==> (1/2/3/4/5)
1
Please choose an option corresponding to the classification of the movie
1. G
2. PG
3. M
4. M15+

Enter your choice ==> (1/2/3/4)
1
Please enter the number of minutes for the movie
60

Please enter the number of copies initially available for the movie
-3
Cannot have 0 or negative copies available
Please enter the number of copies initially available for the movie
0
Cannot have 0 or negative copies available
Please enter the number of copies initially available for the movie
Zero
The number of copies must be a whole number
Please enter the number of copies initially available for the movie
Five
The number of copies must be a whole number
Please enter the number of copies initially available for the movie
4 copies
The number of copies must be a whole number
Please enter the number of copies initially available for the movie
```

## An existing movie in current system (Branch 2)

When an existing movie title in the system is entered, the system will not ask for extra information other than the number of DVDs to add.

### Successful Attempt:

Titanic is already in the system as seen in the previous successful new movie adding section with initially with 3 copies, it is decided that 3 more copies should be added, totalling to 6.

**Titanic** had 3 copies added to it successfully which then adds up to 6 in total.

```
Please enter the title of the new movie to add
Titanic
Please enter the number of copies you would like to add
3
Successfully added 3 copies of Titanic
New total: 6
```

### Unsuccessful Attempts:

#### Unexpected New Movie

Movie names are case sensitive in the community library system so if a movie is entered with a capital letter originally, it must be entered with a capital letter when adding copies. For example as seen below, entering **titanic** asks for the genre to be input when the user is most likely wanting to add movies to the existing version of **Titanic** with a capital 'T'.

```
Please enter the title of the new movie to add
titanic
Please choose the movie genre
1. Drama
2. Action
3. Western
4. History
5. Comedy

Enter your choice ==> (1/2/3/4/5)
```

#### Invalid Number of Copies to Add

Similar to how the initial number of copies is entered, the number of copies to be added must be a positive, whole, numerical number and must not include the 'copies' word at the end. In the example below **four**, **-4**, **0** and **4 copies** were all entered and all do not satisfy the conditions stated.

```
Please enter the title of the new movie to add
Titanic
Please enter the number of copies you would like to add
four
The number of copies must be a whole number
Please enter the number of copies you would like to add
-4
Cannot add 0 or negative dvd copies
Please enter the number of copies you would like to add
0
Cannot add 0 or negative dvd copies
Please enter the number of copies you would like to add
4 copies
The number of copies must be a whole number
Please enter the number of copies you would like to add
```

## 2. Remove DVDs of a movie from the system (Option 2)

### Successful Attempts:

The movie named **Titanic** is in the system already with 5 copies available and 6 total copies. It is decided that 2 copies are to be removed.

**Titanic** had **2** copies removed from it successfully which then is taken to 4 in total with 3 available.

```
Please enter the title of the movie to delete or remove dvds for  
Titanic  
Please enter the number of copies you would like to remove  
2  
Successfully removed 2 copies of Titanic  
New total: 4
```

Then when it is at 4 total copies, it is decided that that the whole movie should be removed from the community library. The one person who was borrowing it returned it as it cannot be deleted with outgoing loans so there are 4 available copies.

**Titanic** had **4** copies removed from it successfully which means that there are no total copies remaining so the movie is completely from the system. If it is required to be added again, the details must be re-entered.

```
Please enter the title of the movie to delete or remove dvds for  
Titanic  
Please enter the number of copies you would like to remove  
4  
Successfully deleted Titanic from the system as it reached 0 copies
```

### Unsuccessful Attempts:

#### Movie doesn't exist

If the movie name corresponding to a movie doesn't exist in the system, it is not possible to remove any DVDs or remove it entirely and is case sensitive. In this example, **Batman** was attempted to be removed, but it doesn't exist in the library and the user is returned to the main menu.

```
Please enter the title of the movie to delete or remove dvds for  
Batman  
Movie doesn't exist  
===== Staff Menu =====  
1. Add new dvds of a new movie to a system  
2. Remove dvds of a movie from the system  
3. Register a new member within the system  
4. Remove a registered member from the system  
5. Display a members contact number given the members name  
6. Display all members who are currently renting a particular movie  
0. Return to main menu
```

#### Invalid Number of Copies to Remove

This works exactly the same as the invalid number of copies to add. In the example below **four**, **-4**, **0** and **4 copies** were all entered and all do not satify the conditions stated in invalid copies to add.

```
Please enter the title of the movie to delete or remove dvds for  
Titanic  
Please enter the number of copies you would like to remove  
four  
The number of copies must be a whole number  
Please enter the number of copies you would like to remove  
-4  
Cannot remove 0 or negative dvd copies  
Please enter the number of copies you would like to remove  
0  
Cannot remove 0 or negative dvd copies  
Please enter the number of copies you would like to remove  
4 copies  
The number of copies must be a whole number  
Please enter the number of copies you would like to remove
```

### **Movie has less available copies than can be removed**

If the movie doesn't have enough available copies that can be removed, it will not work. In this example, the **Titanic**'s copies are at 6 and the available copies are at 5 because a member is currently borrowing it. 6 copies are attempted to be removed but as one of them is not available and on loan to a member so it fails. This member must return their copy before 6 DVDs can be removed.

```
Please enter the title of the movie to delete or remove dvds for  
Titanic  
Please enter the number of copies you would like to remove  
6  
Can not remove more copies than available, there are currently only 5 available copies that can be deleted
```

### 3. Register a new member with the system (Option 3)

The staff member must register a user before they are able to login from the main menu. They must have a valid first name, last name, phone number and pin number.

#### **Successful Attempt:**

A new member with the first name **John**, last name **Smith**, phone number **0406076072** and a pin of **23456** is registered.

```
Please enter the member's first name  
John  
Please enter the member's last name  
Smith  
Please enter the member's phone number  
0406076072  
Please enter the member's pin number  
23456  
***** John Smith 0406076072 23456 *****  
has been successfully added
```

#### **Unsuccessful Attempts:**

### **Whitespace or Empty First Name**

First names cannot be empty or contain only spaces. In this example, the first name was first entered as nothing, and then contained a number of spaces.

```
Please enter the member's first name  
  
Please enter the member's first name  
  
Please enter the member's first name
```

### **Whitespace or Empty Last Name**

Last names just like first names, cannot be empty or contain only spaces. The same inputs as the first name were used on this example.

```
Please enter the member's first name  
John  
Please enter the member's last name  
  
Please enter the member's last name  
  
Please enter the member's last name
```

### Invalid Phone Number

A phone number is only valid when it has exactly 10 digits, the first digit is 0 and each digit is a whole number. In the example below the 1st entered phone number, **041111111a**, is 10 digits and starts with 0 but is invalid as it includes a non-number character. The 2nd phone number, **049999994321** starts with a 0 but is invalid as it has 12 digits. The last phone number, **1411344543** has 10 digits, but is invalid as it doesn't start with 0.

```
Please enter the member's first name  
John  
Please enter the member's last name  
Smith  
  
Please enter the member's phone number  
041111111a  
041111111a is not a valid phone number, please enter a valid phone number  
049999994321  
049999994321 is not a valid phone number, please enter a valid phone number  
1411344543  
1411344543 is not a valid phone number, please enter a valid phone number
```

### Invalid Pin Number

A pin number is only valid when it has 4 to 6 digits and each digit is a whole number. In this example, **123** is invalid as it is under 4 digits, **12344567** is invalid as it is over 6 digits and lastly, **pass** is 4 digits, but is invalid as it doesn't have every digit as a number.

```
Please enter the member's first name  
John  
Please enter the member's last name  
Smith  
Please enter the member's phone number  
0406076072  
  
Please enter the member's pin number  
123  
123 is invalid, please enter a pin between 4 and 6 characters  
12344567  
12344567 is invalid, please enter a pin between 4 and 6 characters  
pass  
pass is invalid, please enter a pin between 4 and 6 characters
```

### 4. Remove a registered member from the system (Option 4)

The system includes a pre-registered member with the first name **John**, last name **Smith**, phone number **0406076072** and a pin of **23456**.

#### Successful Attempt:

In this case **John Smith** is a registered member and has no outstanding borrowings.

```
Please enter the member's first name  
John  
Please enter the member's last name  
Smith  
Member John Smith removed successfully
```

#### Unsuccessful Attempt:

##### Member doesn't exist

After entering both the first and last names of the member, if that member does not exist in the system, it will not work. It is case sensitive so a member registered as **John Smith** is not the same as **john Smith**. In this example a member named **Sarah Smith** was attempted to be removed but they were never registered.

```
Please enter the member's first name  
Sarah  
Please enter the member's last name  
Smith  
Member doesn't exist
```

### **Member is currently borrowing movies**

A member cannot be removed if they are borrowing movies. In this example, **John Smith** is currently borrowing **Titanic** and hasn't returned it yet.

```
Please enter the member's first name  
John  
Please enter the member's last name  
Smith  
Member can not be removed, they still have movies to return
```

### 5. Display a Member's Contact Phone Number (Option 5)

The system includes a pre-registered member with the first name **John**, last name **Smith**, phone number **0406076072** and a pin of **23456**.

#### Successful Attempt:

```
Please enter the member's first name  
John  
Please enter the member's last name  
Smith  
John Smith's contact number:  
0406076072
```

#### Unsuccessful Attempt:

##### **Member doesn't exist**

After entering both the first and last names of the member, if that member does not exist in the system, it will not work. In this example a member named **Sarah Smith** was attempted to have their phone number displayed, but they were never registered.

```
Please enter the member's first name  
Sarah  
Please enter the member's last name  
Smith  
Member doesn't exist
```

### 6. Display Borrowers (Option 6)

In this test case, there are two movies entered into the system:

- **Titanic** with **6 total copies** and of these, **3 are available**.
- **Funnybone** with **10 total copies** and of these, **9 are available**.

There are two members entered as well:

- **John Smith** who is currently borrowing **Titanic** and **Funnybone**
- **Sarah Smith** who is currently borrowing **Titanic**

#### Successful Attempts:

Since both John Smith and Sarah Smith are both borrowing **Titanic**, it displays both:

```
Please enter the title of the movie you want to display the borrowers for  
Titanic  
Current borrowers for Titanic:  
Smith, John  
Smith, Sarah
```

Because only John Smith is borrowing **Funnybone**, it only displays John:

```
Please enter the title of the movie you want to display the borrowers for  
Funnybone  
Current borrowers for Funnybone:  
Smith, John
```

### Unsuccessful Attempt:

If the movie name corresponding to a movie doesn't exist in the system, it is not possible to display the borrowers for it. In this example, **Batman** was attempted to be removed, but it doesn't exist in the library and the user is returned to the main menu.

```
Please enter the title of the movie you want to display the borrowers for  
Batman  
Movie doesnt exist  
===== Staff Menu =====  
1. Add new dvds of a new movie to a system  
2. Remove dvds of a movie from the system  
3. Register a new member within the system  
4. Remove a registered member from the system  
5. Display a members contact number given the members name  
6. Display all members who are currently renting a particular movie  
0. Return to main menu
```

## Member Menu

```
===== Member Menu =====  
1. Browse all the movies  
2. Display all the information about a movie given the title of the movie  
3. Borrow a movie dvd  
4. Return a movie dvd  
5. List current borrowing movie dvds  
6. Display the top 3 movies rented by the members  
0. Return to main menu  
Enter your choice ==> (1/2/3/4/5/6/0)  
-
```

1. Display all information about all the movies DVDs (Option 1)

### Successful Attempts:

All the information about all the available movies in the system with each one's title, genre, classification, duration and available copies sorted alphabetically by movie title are displayed.

In the first test case, there are two movies entered into the system:

- **Titanic** with **10 total copies** and of these, **8 are available**. John Smith and Sarah Smith are both borrowing it.
- **Funnybone** with **10 total copies** and of these, **9 are available**. John Smith is borrowing it.

```
All movies in the community library:  
- Title: Funnybone, Genre: Comedy, Classification: G, Duration: 90 min, Available copies: 9  
- Title: Titanic, Genre: Drama, Classification: PG, Duration: 60 min, Available copies: 8
```

In the other test case, there are no movies in the system:

```
There are no movies in the community library
```

### Unsuccessful Attempts:

It is not possible to unsuccessfully attempt this option.

## 2. Display the information about a specified movie (Option 2)

This option displays the information of a specific movie in the system including its title, genre, classification, duration, and available copies.

### Successful Attempt:

In this example, there **Titanic** is in the only movie in the library with **10 total copies** and of these, **8 are available**. It has the genre **Drama**, classification of **PG** and a duration of **60 minutes**.

When **Titanic** is entered, the information relevant to the member is displayed.

```
Please enter the title of the movie
Titanic
Information for Titanic:
Genre: Drama
Classification: PG
Duration: 60 minutes
Available Copies: 8
```

### Unsuccessful Attempts:

#### Movie does not exist

The movie must exist in the community library for information to be displayed from it the user is forwarded back to the member menu if this is the case. Using the same test data as the successful attempt, two similar cases may arise.

In the first example, **titanic** does not work as it doesn't have a capital letter like in the successful attempt:

```
Please enter the title of the movie
titanic
Movie doesn't exist

===== Member Menu =====

1. Browse all the movies
2. Display all the information about a movie given the title of the movie
3. Borrow a movie dvd
4. Return a movie dvd
5. List current borrowing movie dvds
6. Display the top 3 movies rented by the members
0. Return to main menu

Enter your choice ==> (1/2/3/4/5/6/0)
```

In the other example, **Batman** is entered but again, doesn't exist:

```
Please enter the title of the movie
Batman
Movie doesn't exist

===== Member Menu =====

1. Browse all the movies
2. Display all the information about a movie given the title of the movie
3. Borrow a movie dvd
4. Return a movie dvd
5. List current borrowing movie dvds
6. Display the top 3 movies rented by the members
0. Return to main menu

Enter your choice ==> (1/2/3/4/5/6/0)
```

### 3. Borrow a movie DVD (Option 3)

This option allows a registered member to borrow a movie available in the system.

#### Successful Attempt:

A movie named **Titanic** is currently available and has been borrowed 6 times. The current user logged in named John Smith has no movies currently borrowed. In this example, the user can successfully borrow **Titanic**.

```
Please enter the title of the movie you would like to borrow  
Titanic  
Successfully borrowed Titanic
```

#### Unsuccessful Attempts:

##### Movie does not exist

If the movie name corresponding to a movie doesn't exist in the system, it is not possible for a member to borrow that movie. In this example, the user attempts to borrow **Batman**, and this movie is not in the system. As always, this is case sensitive.

```
Please enter the title of the movie you would like to borrow  
Batman  
Movie doesn't exist  
  
===== Member Menu =====  
1. Browse all the movies  
2. Display all the information about a movie given the title of the movie  
3. Borrow a movie dvd  
4. Return a movie dvd  
5. List current borrowing movie dvds  
6. Display the top 3 movies rented by the members  
0. Return to main menu
```

##### User already borrowing the movie

If the user is already borrowing a movie, this same movie cannot be borrowed again until returned.

In this example, the user is already borrowing **Titanic** but attempts to borrow it again.

```
Please enter the title of the movie you would like to borrow  
Titanic  
Can not borrow more than one copy of a movie at a time
```

##### User is borrowing 5 movies already

Each user is only allowed to borrow 5 movies at a time. In this example, the user is borrowing **5** movies already and attempts to borrow a 6th. The error occurs before they can enter a title.

```
Can not borrow more than 5 movies at a time
```

##### There are 10 borrowers of a movie already

A movie is only allowed to have 10 borrowers at any given time. In this example, the movie **Titanic** has **10 current borrowers** and the logged in **user is not currently borrowing it**.

```
Please enter the title of the movie you would like to borrow  
Titanic  
Can not have more than 10 borrowers of a movie at a time
```

### 4. Return a movie DVD (Option 4)

This option allows a registered member to return a movie in the system that they are currently borrowing.

#### Successful Attempt:

In this example, the current logged in member has successfully returned the movie **Titanic** which they were borrowing.

```
Please enter the name of the movie to return  
Titanic  
Successfully returned Titanic
```

Unsuccessful Attempts:

#### Movie Doesn't Exist

In the same way that you can't borrow a movie that doesn't exist, you can't return one. In this case **Batman** was attempted to be returned.

```
Please enter the name of the movie to return  
Batman  
Movie doesn't exist
```

#### User is not Borrowing the DVD specified

If a user is not borrowing a movie but it exists, they are not able to return it. In this case the movie **Funnybone** exists but the **current user** named John Smith **is not borrowing** it and has returned it already.

```
Please enter the name of the movie to return  
Funnybone  
You aren't currently borrowing this movie
```

### 5. List currently borrowed DVDs by the current logged in member (Option 5)

Allows a registered member to display all of their currently borrowed DVD's in the library.

Successful Attempts:

In this example the user **John Smith** is borrowing both **Titanic** and **Funnybone** which account for **2** out of the **5** maximum they can borrow.

```
Currently borrowing 2/5 movies  
Your currently borrowed dvds:  
- Title: Funnybone, Genre: Comedy, Classification: G, Duration: 90 min, Available copies: 9  
- Title: Titanic, Genre: Drama, Classification: PG, Duration: 60 min, Available copies: 8
```

In this next example, **Sarah Smith** is not borrowing any movies which is **0** out of the maximum of **5**.

```
Currently borrowing 0/5 movies  
You are not borrowing any DVDs
```

Unsuccessful Attempts:

It is not possible to unsuccessfully attempt this option.

### 6. Display top 3 movies that have been borrowed the most (Option 6)

Allows a registered member to display the information of the top 3 rented movies of all time in the community library.

Successful Attempts:

In this example, user **John Smith** is currently borrowing both **Titanic** and **Funnybone**. Along with John, **Sarah Smith** is currently borrowing only **Titanic**. There have been **2** members in the past who have borrowed and returned **Funnybone** as well. This puts Funnybone in first place at 3 total borrows, Titanic in second, and no movie in third as only 2 movies are in the library at this current time.

```
Top 3 movies:  
1st: Funnybone - 3  
2nd: Titanic - 2  
3rd: nil
```

Carrying on from the previous example, **John Smith** along with **Sarah Smith** decide to borrow **CAB301 Lecture Marathon**. When the top 3 movies option is picked again it displays this:

```
Top 3 movies:  
1st: Funnybone - 3  
2nd: CAB301 Lecture Marathon - 2  
3rd: Titanic - 2
```

When someone returns a movie the numbers do not decrease as they are the all-time borrowings but when a movie is deleted these numbers are removed. For example, following on from the previous case, everyone returns **Titanic** and it has all of its copies removed so it is deleted. This is what the list then looks like:

```
Top 3 movies:  
1st: Funnybone - 3  
2nd: CAB301 Lecture Marathon - 2  
3rd: nil
```

Following on from the previous case again, both **Funnybone** and **CAB301 Lecture Marathon** both have all their copies removed after everyone returns them. This leaves the system with no movies with any borrowing history.

```
Top 3 movies:  
1st: nil  
2nd: nil  
3rd: nil
```

Keeping the same library as after the previous example, a new movie named **Igloo Raiders** is added and nobody is borrowing it. The list is unchanged as it only shows movies that have at least 1 borrow.

```
Top 3 movies:  
1st: nil  
2nd: nil  
3rd: nil
```

#### Unsuccessful Attempts:

It is not possible to unsuccessfully attempt this option.