Queensland University of Technology

School of Information Systems

Science and Engineering Faculty

IFN 664 Advanced Algorithms and Computational Complexity

Development of a Software Application for a Community Library to Manage Its Movie DVDs

Name: Ian Choi

Student ID: N10421106

Lecturer: Maolin Tang

Contents

Introduction

The goal of this project is to develop a C# based console software application for a community library to manage its movie DVDs. There are staff and member menu in this application which can be selected in the main menu. Both menu can be accessed with the authentication. The staff menu is basically used to manage movie, DVDs, and member in the system. On the other hand, the member menu is created to function to borrow, return, and check information of the movie. In addition, a function to identify the top 3 movies which are most frequently borrowed by the members is developed using bubble sort. Lastly, the hash table data structure is used to store the movie data.

Design and Analysis of Algorithms

To display top three most frequently borrowed movie DVDs in descending order by the number of times that the movie DVDs have been borrowed, the bubble sort has been used in the algorithm. First, sort the whole movie array by BorrowCount. Then, to check the array is sorted, give isSorted value of true. Next, compare the elements(movies) in the array from 0 to -1. Then, move the movie with the smaller BorrowCount between i and i+1. If there happens a switch, give isSorted value of false. Lastly, if the isSorted is false, it means the loop must continue because the array needs to be checked if it is fully sorted. Otherwise, if isSorted is true, it means the switch did not happen which concludes that the array is fully sorted and escape the loop.

Pseudo Code:

```
                              --- Sort Algorithm ---


       void displayTop3( )
           movies = _movies.toArray( );
           bool isSorted;
           do
           {   isSorted = true;
               for ( i = 0; I < movies.Length – 1; i++ )
               {
                   if ( ( movies[ i ] as Movie).BorrowCount < ( movies [ i + 1 ] as
                   Movie ). BorrowCount)
                       {
                               IMovie tmp = movies [ i ];
                               movies [ i ] = movies [ i + 1 ];
                               movies [ i + 1 ] = tmp;
           isSorted = false;
        }
      }
    }
} while ( !isSorted );

            --- Return TOP 3 ---

for ( var i = 0; i < 3 && i < movies.Length; i++ )
    return [{ i + 1 } ] { movies[ i ].Title }
```

The time efficiency or complexity of the algorithm used, bubble sort, is **O(n^2)** because the comparison occurs n^2 times in worst case. In bubble sort, the outer loop repeats n-1 times and inner loop repeats n-1-I times. Therefore, the total comparison occurs (n-1) (n-1-i) times which leads to O(n^2).

Hash Table

Since the hash table data structure is considered as efficient method to store the movie data, the system generic library of C# is used to create hash table. Hash table stores data to the key dictionally that has O(n) of time complexity in the worst case.

```
public MovieCollection( )
    {
        _movies = new Hashtable();
    }
```

The following code is created to add data to the hash table.

```
void add( IMovie aMovie )
{
  _movies.Add( aMovie.Title.ToUpper(), aMovie );
}
```

The following code is to delete data in the hash table.

```
void delete( IMovie aMovie )
{
  _movies.Remove( aMovie.Title.ToUpper());
}
```

The following code is to search in the hash table with movie title.

```
        public IMovie search(IMovie aMovie)
        {
            if(_movies.ContainsKey(aMovie.Title.ToUpper()))
            return _movies[aMovie.Title.ToUpper()] as IMovie;
            return null;
        }
```

The following code is to remove all the data in the hash table.

```
        void clear()
        {
            _movies.Clear();
        }
```

Test Plan and Test Results

The test plan for this console application is based on the required functions in the project specification.

The software application should have a staff menu that allows the staff to do the following:

- Add DVDs of a new movie to the system.
- Add DVDs of an existing movie to the system.
- Remove DVDs of a movie from the system – if all the DVDs of a movie are removed, the movie should also be removed from the system.
- Register a new member with the system. When a member is being registered via a staff member, a four-digit password is set for the member.
- Remove a registered member from the system.
- Find a member's contact phone number, given the member's full name.
- Find all the members who are currently renting a particular movie.

The software application should have a member menu that allows a registered member to do the following:

- Display all the information about all the movie DVDs in dictionary order of the movie title, including the number of the movie DVDs currently in the community library.
- Display the information about a movie, given the title of the movie.
- Borrow a movie DVD from the community library, given the title of the movie DVD.
- Return a movie DVD to the community library, given the title of the movie DVD.
- List current movie DVDs that are currently borrowing by the registered member.
- Display the top three (3) most frequently borrowed movie DVDs by the members in the

descending order of their frequency. The display should include the title of the movie and frequency (the number of times that the movie DVDs have been borrowed by registered members by now).

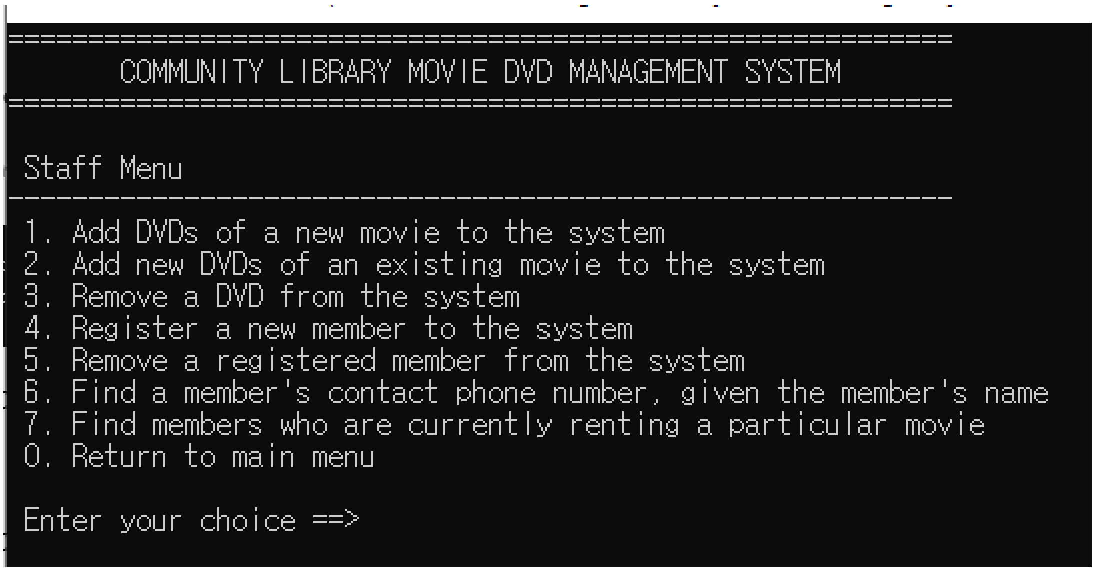
The application is designed and developed to achieve these tasks successfully and manually tested with the manually generated member and movie data while testing the application. The specification which is not included in the list above is also tested. The other test plan is as followed:

- Display error when there is no copy to borrow.
- Display error when the member tries to borrow more than 5 DVDs at a time.

The result of the testing turned out to successful by passing all of the testing specifications. The following images are the screenshots of successful testing.

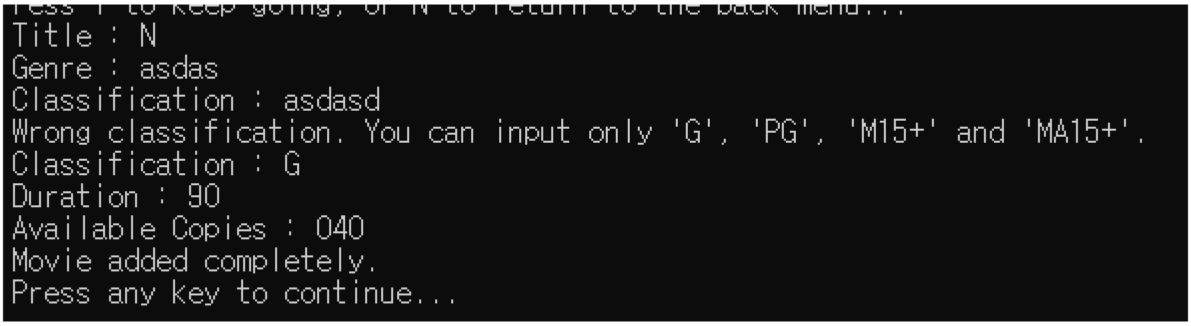
The software application should have a staff menu that allows the staff to do the following:

Staff menu

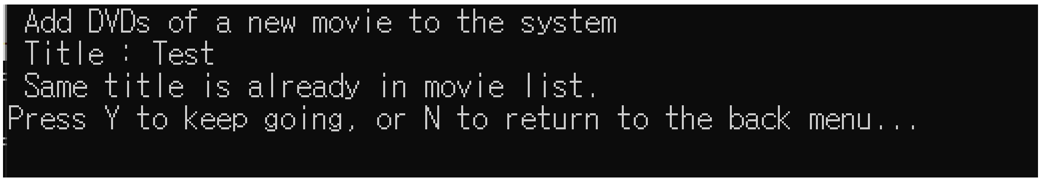


```
============================================================
       COMMUNITY LIBRARY MOVIE DVD MANAGEMENT SYSTEM
============================================================

 Staff Menu
------------------------------------------------------------
 1. Add DVDs of a new movie to the system
 2. Add new DVDs of an existing movie to the system
 3. Remove a DVD from the system
 4. Register a new member to the system
 5. Remove a registered member from the system
 6. Find a member's contact phone number, given the member's name
 7. Find members who are currently renting a particular movie
 0. Return to main menu

 Enter your choice ==>
```

Add DVDs of a new movie to the system.

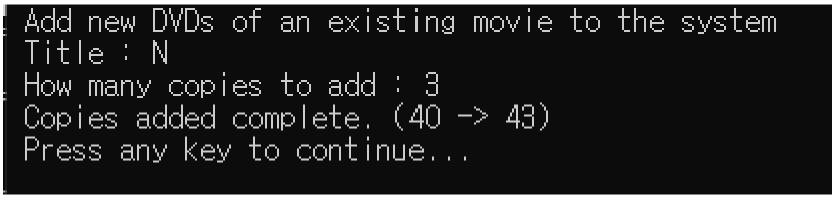


```
ress Y to keep going, or N to return to the back menu...
Title : N
Genre : asdas
Classification : asdasd
Wrong classification. You can input only 'G', 'PG', 'M15+' and 'MA15+'.
Classification : G
Duration : 90
Available Copies : 040
Movie added completely.
Press any key to continue...
```



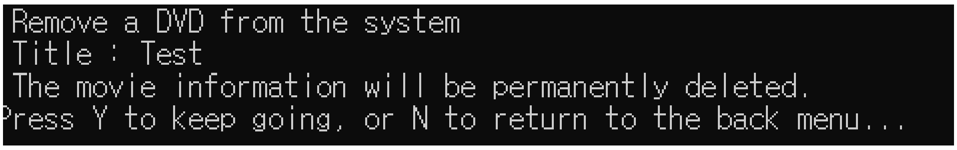
```
 Add DVDs of a new movie to the system
 Title : Test
 Same title is already in movie list.
Press Y to keep going, or N to return to the back menu...
```

Add DVDs of an existing movie to the system.



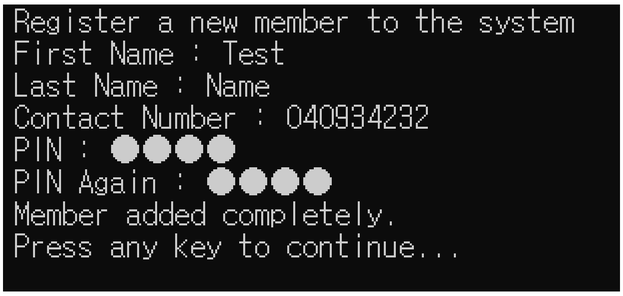
```
 Add new DVDs of an existing movie to the system
 Title : N
 How many copies to add : 3
 Copies added complete. (40 -> 43)
 Press any key to continue...
```

Remove DVDs of a movie from the system – if all the DVDs of a movie are removed, the movie should also be removed from the system.



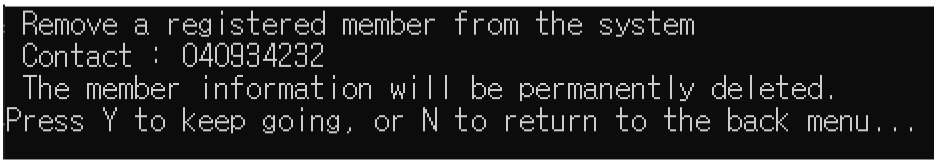
```
 Remove a DVD from the system
 Title : Test
 The movie information will be permanently deleted.
Press Y to keep going, or N to return to the back menu...
```

Register a new member with the system. When a member is being registered via a staff member, a four-digit password is set for the member.



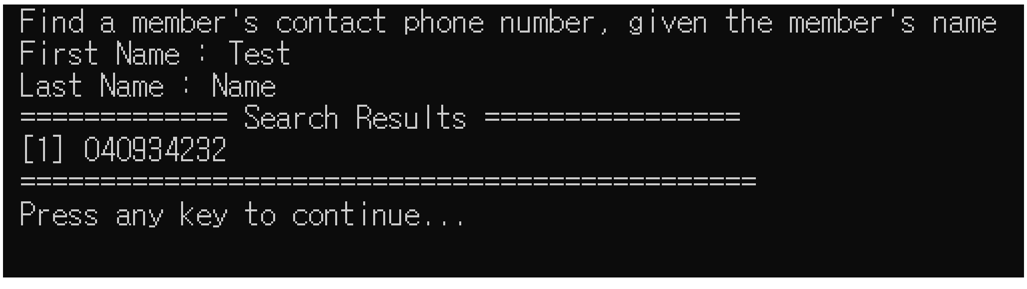
```
Register a new member to the system
First Name : Test
Last Name : Name
Contact Number : 040934232
PIN : ●●●●
PIN Again : ●●●●
Member added completely.
Press any key to continue...
```

Remove a registered member from the system.



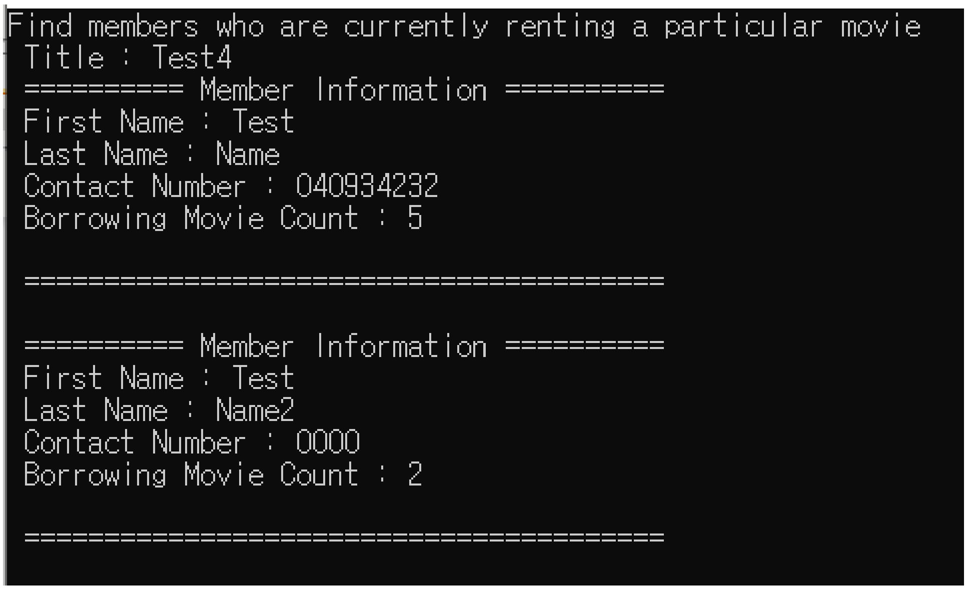
```
 Remove a registered member from the system
 Contact : 040934232
 The member information will be permanently deleted.
Press Y to keep going, or N to return to the back menu...
```

Find a member’s contact phone number, given the member’s full name.



```
Find a member's contact phone number, given the member's name
First Name : Test
Last Name : Name
============= Search Results ================
[1] 040934232
=============================================
Press any key to continue...
```

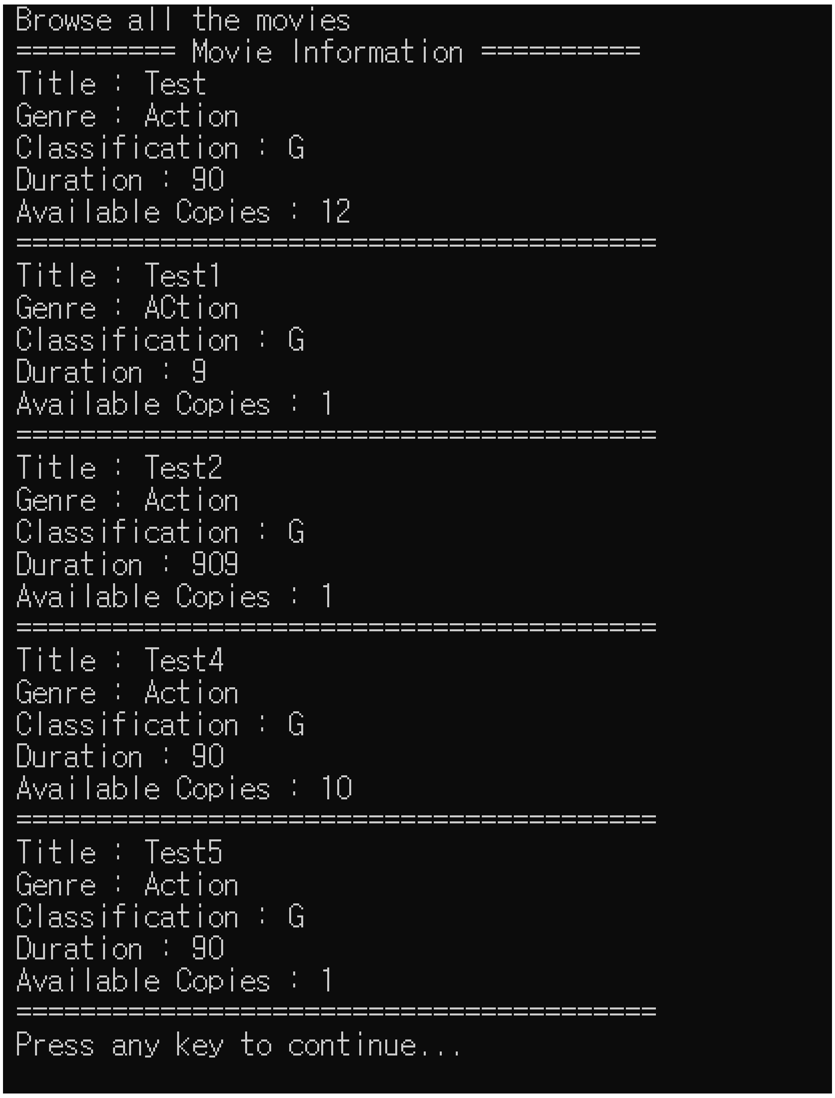
Find all the members who are currently renting a particular movie.



```
Find members who are currently renting a particular movie
 Title : Test4
 ========== Member Information ==========
 First Name : Test
 Last Name : Name
 Contact Number : 040934232
 Borrowing Movie Count : 5

 ========================================

 ========== Member Information ==========
 First Name : Test
 Last Name : Name2
 Contact Number : 0000
 Borrowing Movie Count : 2

 ========================================
```

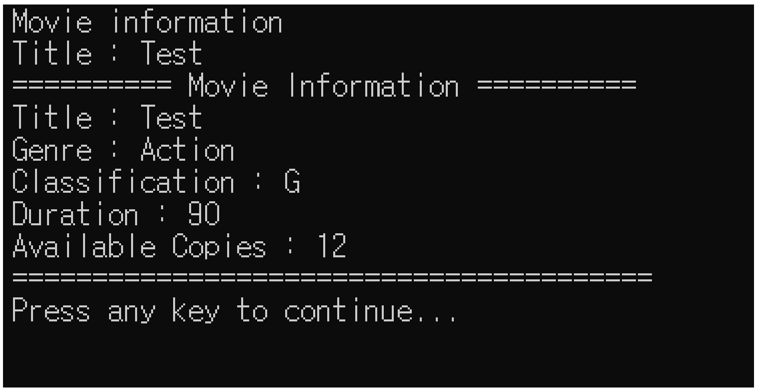
The software application should have a member menu that allows a registered member to do the following:

Display all the information about all the movie DVDs in dictionary order of the movie title, including the number of the movie DVDs currently in the community library.



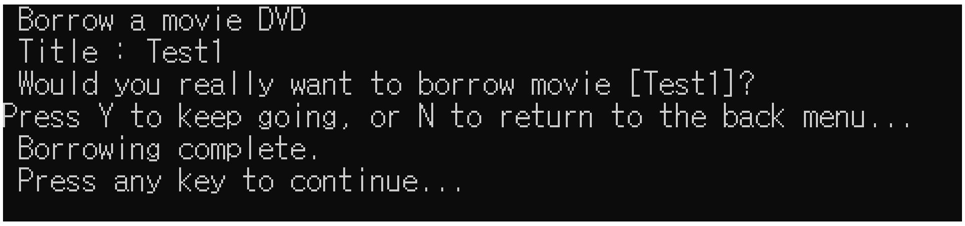
```
Browse all the movies
========== Movie Information ==========
Title : Test
Genre : Action
Classification : G
Duration : 90
Available Copies : 12
========================================
Title : Test1
Genre : ACtion
Classification : G
Duration : 9
Available Copies : 1
========================================
Title : Test2
Genre : Action
Classification : G
Duration : 909
Available Copies : 1
========================================
Title : Test4
Genre : Action
Classification : G
Duration : 90
Available Copies : 10
========================================
Title : Test5
Genre : Action
Classification : G
Duration : 90
Available Copies : 1
========================================
Press any key to continue...
```

Display the information about a movie, given the title of the movie.



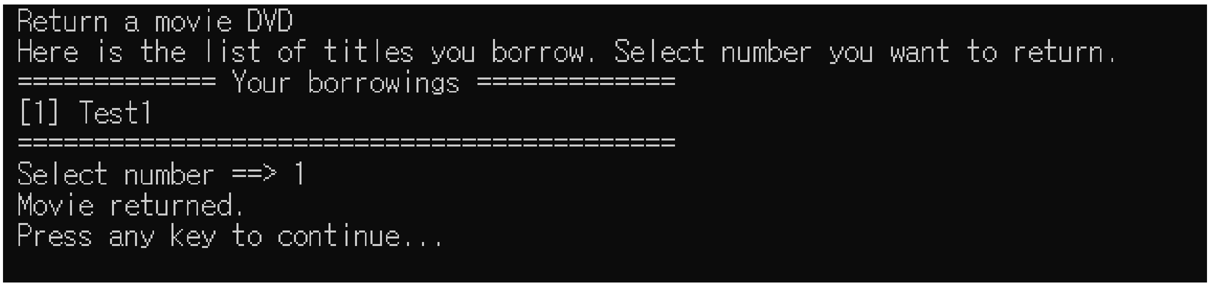
```
Movie information
Title : Test
========== Movie Information ==========
Title : Test
Genre : Action
Classification : G
Duration : 90
Available Copies : 12
========================================
Press any key to continue...
```

Borrow a movie DVD from the community library, given the title of the movie DVD.



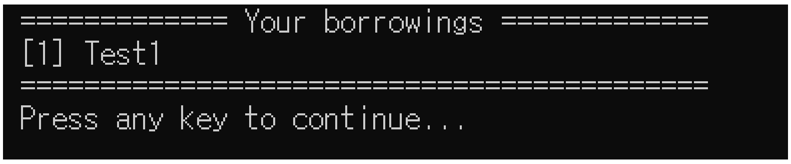
```
 Borrow a movie DVD
 Title : Test1
 Would you really want to borrow movie [Test1]?
Press Y to keep going, or N to return to the back menu...
 Borrowing complete.
 Press any key to continue...
```

Return a movie DVD to the community library, given the title of the movie DVD.



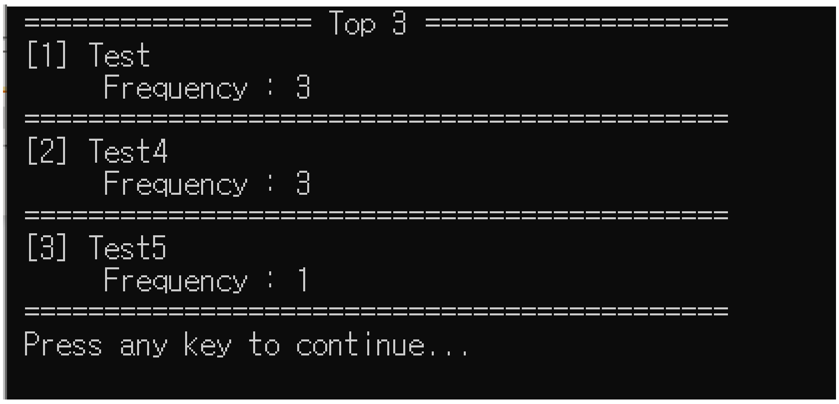
```
Return a movie DVD
Here is the list of titles you borrow. Select number you want to return.
============= Your borrowings =============
[1] Test1
===========================================
Select number ==> 1
Movie returned.
Press any key to continue...
```

List current movie DVDs that are currently borrowing by the registered member.



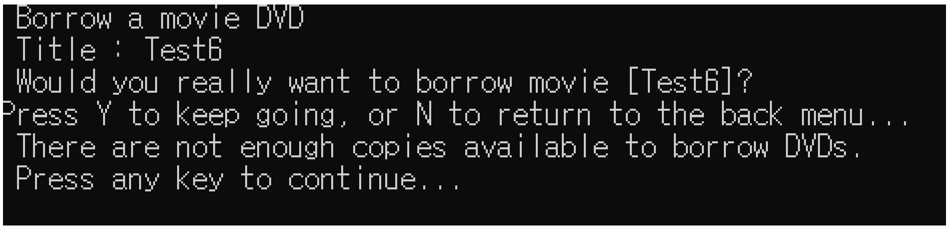
```
============= Your borrowings =============
[1] Test1
===========================================
Press any key to continue...
```

Display the top three (3) most frequently borrowed movie DVDs by the members in the descending order of their frequency. The display should include the title of the movie and frequency (the number of times that the movie DVDs have been borrowed by registered members by now).



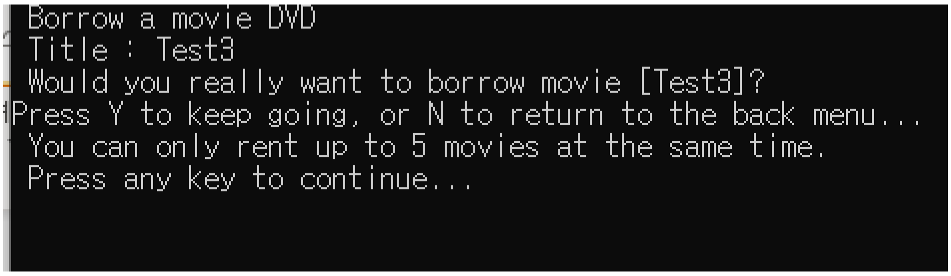
```
================== Top 3 ===================
[1] Test
     Frequency : 3
============================================
[2] Test4
     Frequency : 3
============================================
[3] Test5
     Frequency : 1
============================================
Press any key to continue...
```

Display error when there is no copy to borrow.



```
 Borrow a movie DVD
 Title : Test6
 Would you really want to borrow movie [Test6]?
Press Y to keep going, or N to return to the back menu...
 There are not enough copies available to borrow DVDs.
 Press any key to continue...
```

Display error when the member tries to borrow more than 5 DVDs at a time.



```
 Borrow a movie DVD
 Title : Test3
 Would you really want to borrow movie [Test3]?
Press Y to keep going, or N to return to the back menu...
 You can only rent up to 5 movies at the same time.
 Press any key to continue...
```