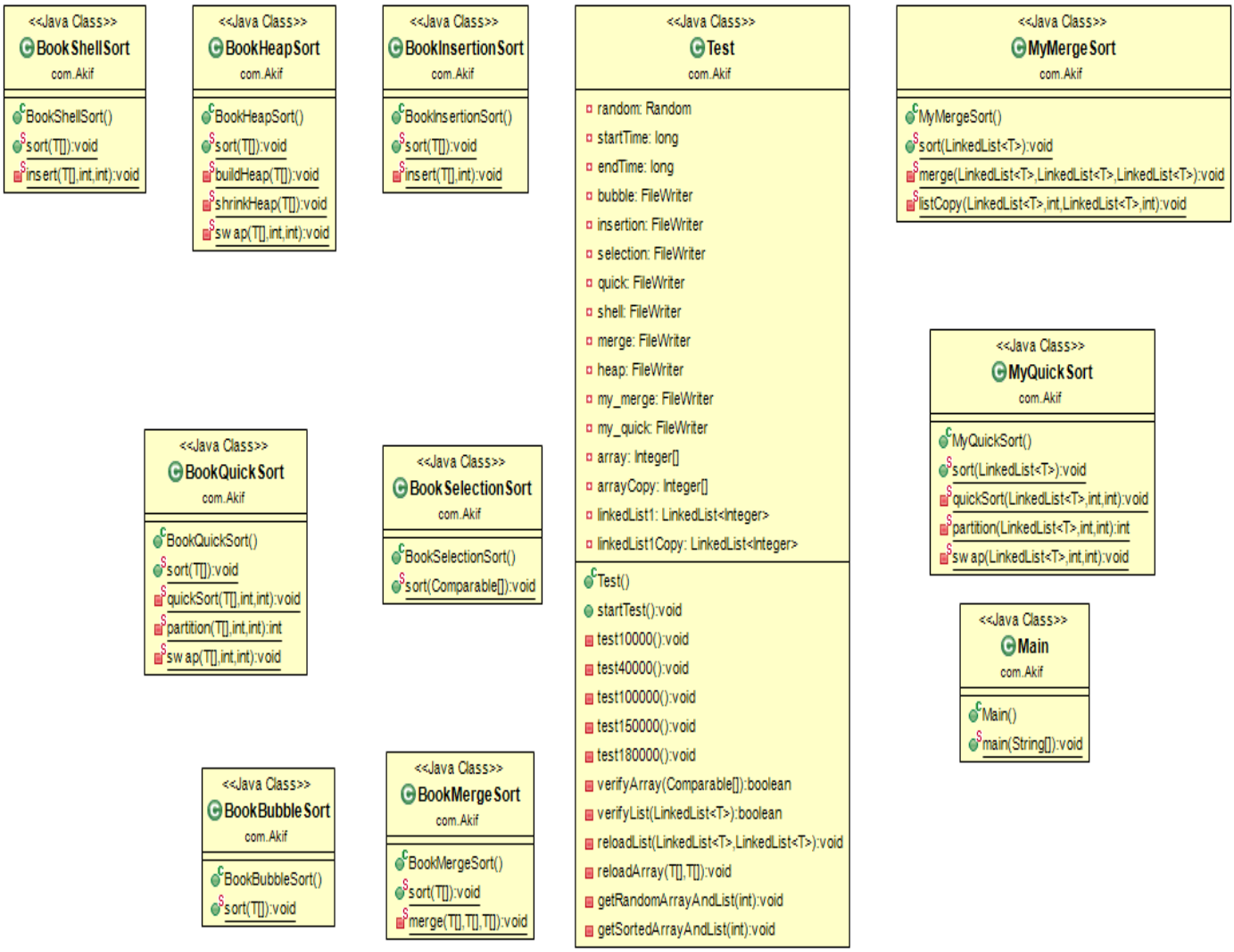
GIT Department of Computer Engineering CSE 222/505 - Spring 2020 Homework 6 Report

Akif KARTAL
171044098

Q2 REPORT

1. CLASS DIAGRAMS



2. PROBLEM SOLUTION APPROACH

My Problem solution steps are;

– Specify the problem requirements
– Analyze the problem
– Design an algorithm and Program
– Implement the algorithm
– Test and verify the program
– Maintain and update the program

1) **Specify the problem requirements :** I understand the problem.
2) **Analyze the problem :** I identify;
   – Input data
   – Output data
   – Additional requirements and constraints

3) **Design an algorithm and Program :** I divide the problem into sub-problems. I listed major steps (sub-problems). I break down each step into a more detailed list. To do these We have to divide this big project into small pieces.

➡ **Implement the algorithm :** I wrote the algorithm in Java by converting each step into statements of Java (classes ,methods etc.)

Firstly, I wrote **7 algorithms** from book. After, I wrote the Quick Sort and Merge Sort algorithms such that data is stored in a **linked list**.

Then, I start to write test class to compare algorithms with arrays and list that has different size to compare running time.

To test Selection Sort, Bubble Sort, Insertion Sort, Shell Sort, Merge Sort, Heap Sort and My Merge sort algorithm I apply this process;

- For length of **ten thousand (10000)** I created **20 random** array or list and **1 sorted** array or list I save running time to a file then I calculate the average of them.

- For length of **forty thousand (40000)** I created **20 random** array or list and **1 sorted** array or list I save running time to a file then I calculate the average of them.

- For length of **one hundred thousand (100000)** I created **20 random** array or list and **1 sorted** array or list I save running time to a file then I calculate the average of them.

- For length of **one hundred and fifty thousand (150000)** I created **1 random** array or list and **1 sorted** array or list I save running time to a file then I calculate the average of them. In this size since waiting time is to long about 2 hours, I used 1 random array instead 20.

- For length of **one hundred and eighty thousand (180000).** I created **1 random** array or list and **1 sorted** array or list I save running time to a file then I calculate the average of them. In this size since waiting time is to long about 5 hours, I used 1 random array instead 20.

To test **Quick and My Quick sort** algorithm I used 2 random array since they take too many time. I didn't use sorted array since I got **stackoverflow exception**.

Lastly, I draw a **graph** by using these values.

4) **Test and verify the program:** To test program I wrote the **Test** class in this class, I created random and sorted array for each size and sort them after sorting I checked them **whether sorted or not** if sorting algorithm is fail then I repeat them if it is successful I **returned back array old form(not sorted)** to use same array in other algorithms.
5) **Maintain and update the program :** I keep the program up-to-date

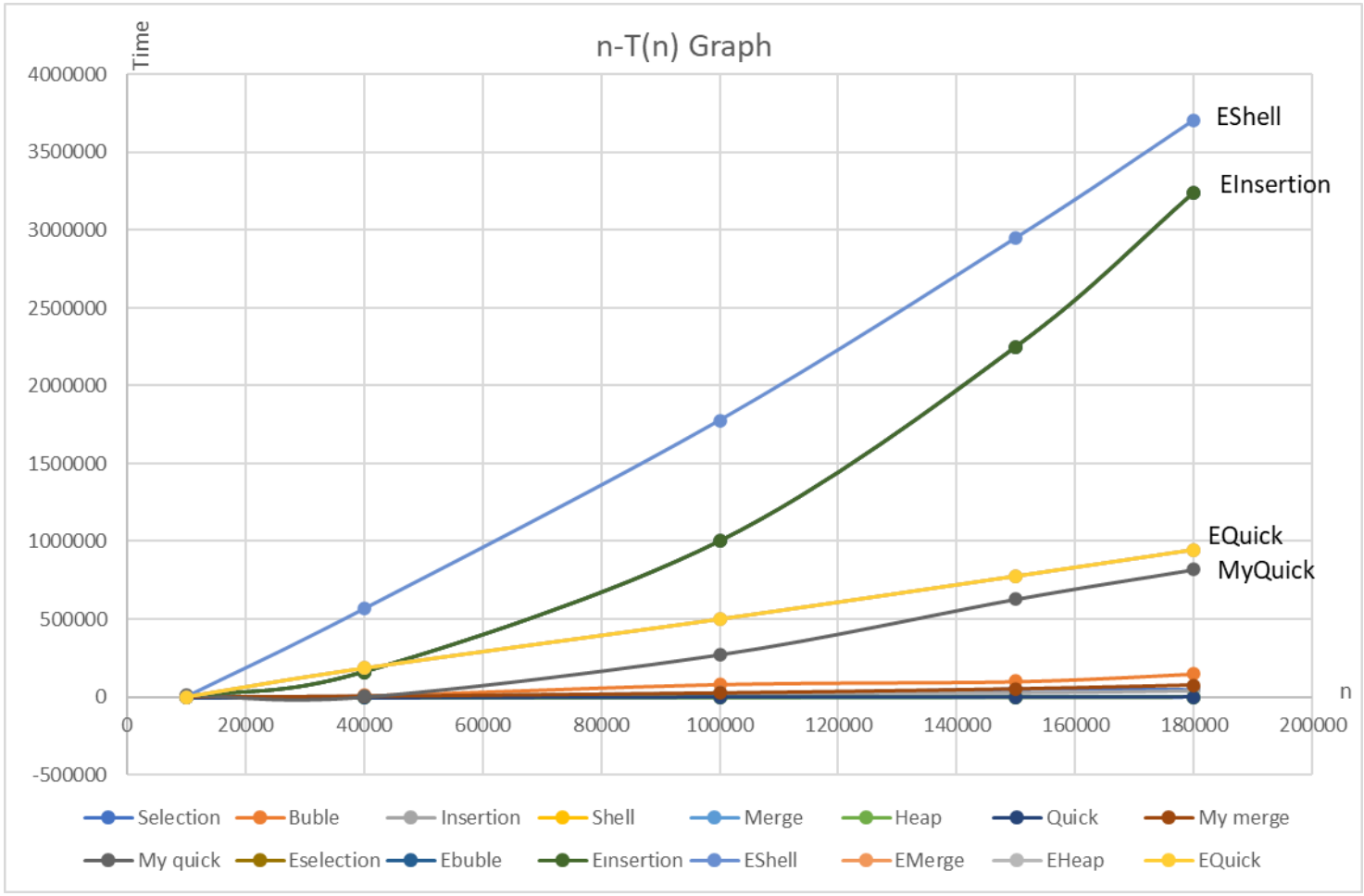
3. TEST CASES

Test ID	Test Case	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
T1	Test Selection Sort	1) Create Random and Sorted Arrays. 2) Call Method. 3) Verify array is sorted or not.	100 Random 5 Sorted array	The test is successful	As Expected	**Pass**
T2	Test Bubble Sort	1) Create Random and Sorted Arrays. 2) Call Method. 3) Verify array is sorted or not.	100 Random 5 Sorted array	The test is successful	As Expected	**Pass**
T3	Test Insertion Sort	1) Create Random and Sorted Arrays. 2) Call Method. 3) Verify array is sorted or not.	100 Random 5 Sorted array	The test is successful	As Expected	**Pass**
T4	Test Shell Sort	1) Create Random and Sorted Arrays. 2) Call Method. 3) Verify array is sorted or not.	100 Random 5 Sorted array	The test is successful	As Expected	**Pass**
T5	Test Merge Sort	1) Create Random and Sorted Arrays. 2) Call Method. 3) Verify array is sorted or not.	100 Random 5 Sorted array	The test is successful	As Expected	**Pass**
T6	Test Heap Sort	1) Create Random and Sorted Arrays. 2) Call Method. 3) Verify array is sorted or not.	100 Random 5 Sorted array	The test is successful	As Expected	**Pass**
T7	Test Quick Sort	1) Create Random and Sorted Arrays. 2) Call Method. 3) Verify array is sorted or not.	100 Random 5 Sorted array	The test is successful	As Expected	**Pass**
T8	Test My Merge Sort	1) Create Random and Sorted Arrays. 2) Call Method. 3) Verify array is sorted or not.	100 Random 5 Sorted array	The test is successful	As Expected	**Pass**
T9	Test My Quick Sort	1) Create Random and Sorted Arrays. 2) Call Method. 3) Verify array is sorted or not.	100 Random 5 Sorted array	The test is successful	As Expected	**Pass**

4. RUNNING AND RESULTS

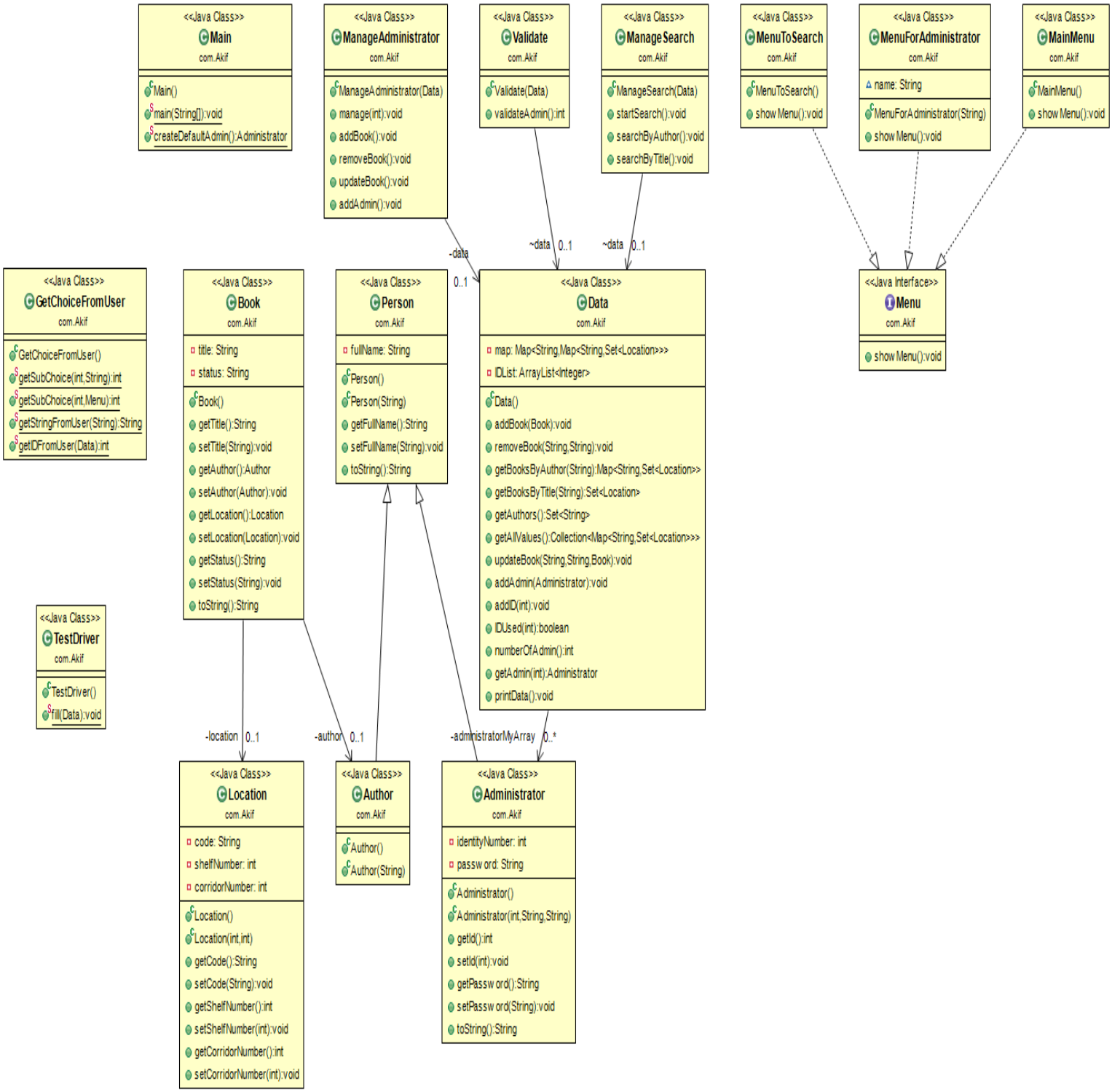
Test ID	Test Result
T1	--- ***Selection Sort*** Book Selection sort time is 431 ms Book Selection sort is successful (true/false): true ---
T2	--- ***Bubble Sort*** Book Bubble sort time is 1358 ms Book Bubble sort is successful (true/false): true ---
T3	--- ***Insertion Sort*** Book Insertion sort time is 458 ms Book Insertion sort is successful (true/false): true ---
T4	--- ***Shell Sort*** Book Shell sort time is 27 ms Book Shell sort is successful (true/false): true ---
T5	--- ***Merge Sort*** Book Merge sort time is 13 ms Book Merge sort is successful (true/false): true ---
T6	--- ***Heap Sort*** Book Heap sort time is 31 ms Book Heap sort is successful (true/false): true ---
T7	--- ***Quick Sort*** Book Quick sort time is 17 ms Book Quick sort is successful (true/false): true ---

T8	---------- ***My Merge Sort*** My Merge sort time is 607ms My Merge sort is successful (true/false): true ----------
T9	---------- ***My Quick Sort*** My Quick sort time is 3349ms My Quick sort is successful (true/false): true ----------



Q3 REPORT

1. CLASS DIAGRAMS



2. PROBLEM SOLUTION APPROACH

My Problem solution steps are;

– Specify the problem requirements
– Analyze the problem
– Design an algorithm and Program
– Implement the algorithm
– Test and verify the program
– Maintain and update the program

1) **Specify the problem requirements :** I understand the problem.
2) **Analyze the problem :** I identify;
   – Input data
   – Output data
   – Additional requirements and constraints

3) **Design an algorithm and Program :** I divide the problem into sub-problems. I listed major steps (sub-problems). I break down each step into a more detailed list. To do these We have to divide this big project into small pieces.

⟹ **Implement the algorithm :** I wrote the algorithm in Java by converting each step into statements of Java (classes ,methods etc.)

Firstly, I wrote the **Data** class to keep all data about the system. In this class I stored the information about the books **in nested map** and I write operation about the data in this class. Then I wrote all the classes about the system such as Book, Author, Administrator and Menu classes.

To perform admin operations I wrote the **ManageAdministrator** class. In this class I wrote operations about the admin of System.

To perform search operations I wrote the **ManageSearch** class. In this class I wrote operations about the search for user.

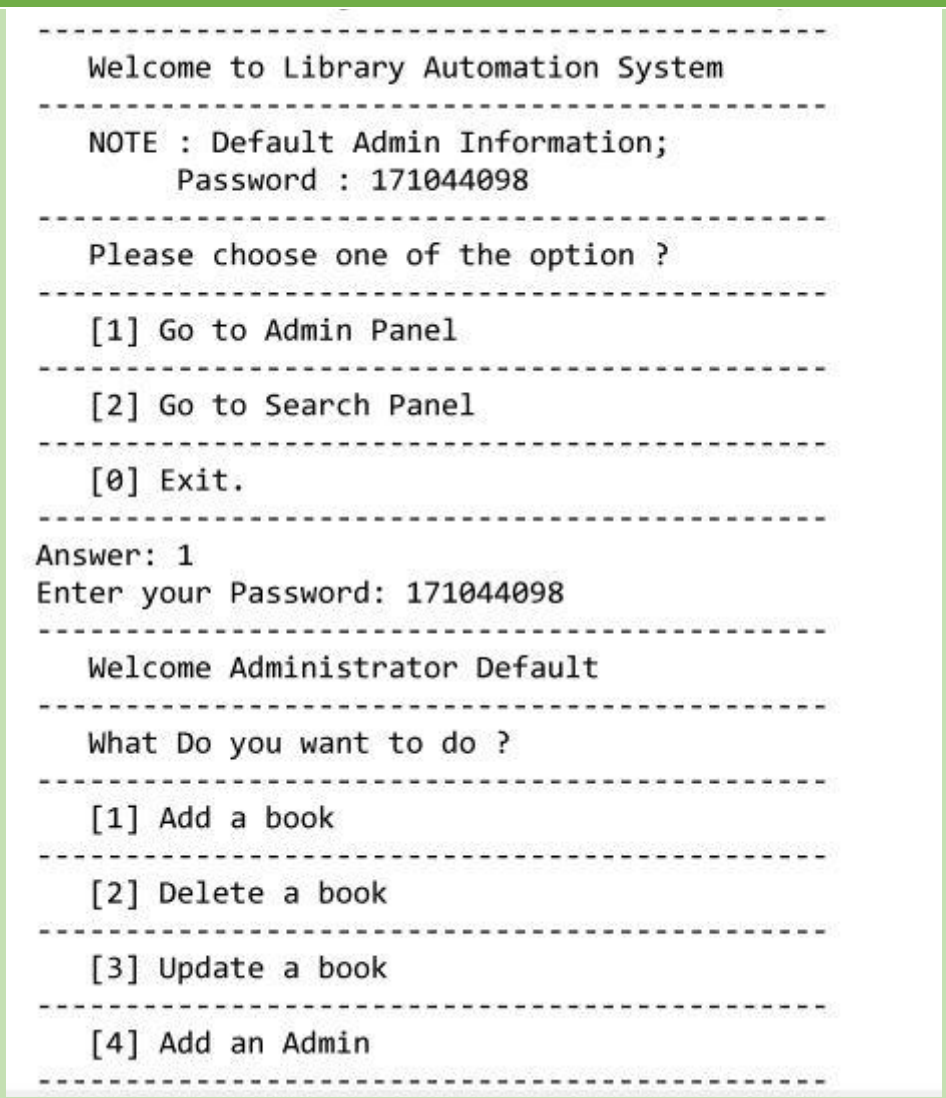
4) **Test and verify the program:** To test program I wrote the **TestDriver** class. In this class I fill the books(map) from a file after filling operation in **Main class in main** method I showed the main menu to the user I did operations based on user's choice.
5) **Maintain and update the program :** I keep the program up-to-date.

3. TEST CASES

Test ID	Test Case	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
T1	Test Admin Login	1) Run program 2) Go to Admin Panel 3) Enter Password	“171044098”	Admin should Login into Application.	As Expected	Pass
T2	Test Add a book	1) Run program 2) Go to Admin Panel 3) Enter Password 4) Enter 1 5) Enter Information	Leo Tolstoy War and Peace 5 1	The book has added.	As Expected	Pass
T3	Test Delete a book	1) Run program 2) Go to Admin Panel 3) Enter Password 4) Enter 2 5) Enter Information	William Shakespeare Hamlet	The book has deleted.	As Expected	Pass
T4	Test Update a book	1) Run program 2) Go to Admin Panel 3) Enter Password 4) Enter 3 5) Enter Information	William Shakespeare Hamlet William Shakespeare Hamlet2 2 4	The book has updated.	As Expected	Pass
T5	Test Add an Admin	1) Run program 2) Go to Admin Panel 3) Enter Password 4) Enter 4 5) Enter Information	akif kartal 123 456	The admin has added.	As Expected	Pass
T6	Test Print all Data	1) Run program 2) Go to Admin Panel 3) Enter Password 4) Enter 5 5) Enter Information		All Data has printed in a good shape.	As Expected	Pass

T7	Search by author name	1) Run program 2) Go to Search Panel 3) Enter 1 4) Enter Information	William Shakespeare	All books of the author in the library will be listed on the screen. Then, whichever book the user chooses, the location(s) of that book will be displayed.	As Expected	**Pass**
T8	Search by book title	1) Run program 2) Go to Search Panel 4) Enter 2 5) Enter Information	To Kill a Mockingbird	Author name, location and status will be shown.	As Expected	**Pass**

4. RUNNING AND RESULTS

Test ID	Test Result
T1	

T2

```
-------------------------------------------
   Welcome Administrator Default
-------------------------------------------
   What Do you want to do ?
-------------------------------------------
   [1] Add a book
-------------------------------------------
   [2] Delete a book
-------------------------------------------
   [3] Update a book
-------------------------------------------
   [4] Add an Admin
-------------------------------------------
   [5] Print all Data
-------------------------------------------
   [0] Main Menu.
-------------------------------------------
Answer: 1
Enter author full name: Leo Tolstoy
Enter title of book: War and Peace
Enter corridor number(0-5): 5
Enter shelf number(0-4): 1
-------------------------------------------
   Welcome Administrator Default
-------------------------------------------
   What Do you want to do ?
-------------------------------------------
   [1] Add a book
-------------------------------------------
   [2] Delete a book
-------------------------------------------
   [3] Update a book
```

T3

```
-------------------------------------------
   Welcome Administrator Default
-------------------------------------------
   What Do you want to do ?
-------------------------------------------
   [1] Add a book
-------------------------------------------
   [2] Delete a book
-------------------------------------------
   [3] Update a book
-------------------------------------------
   [4] Add an Admin
-------------------------------------------
   [5] Print all Data
-------------------------------------------
   [0] Main Menu.
-------------------------------------------
Answer: 2
Enter author Name of Book: William Shakespeare
Enter Book Name: Hamlet
1 book removed from library
-------------------------------------------
   Welcome Administrator Default
-------------------------------------------
   What Do you want to do ?
-------------------------------------------
   [1] Add a book
-------------------------------------------
   [2] Delete a book
-------------------------------------------
   [3] Update a book
-------------------------------------------
```

T4

```
-------------------------------------------
   Welcome Administrator Default
-------------------------------------------
   What Do you want to do ?
-------------------------------------------
   [1] Add a book
-------------------------------------------
   [2] Delete a book
-------------------------------------------
   [3] Update a book
-------------------------------------------
   [4] Add an Admin
```


File - Main

```
-------------------------------------------
   [5] Print all Data
-------------------------------------------
   [0] Main Menu.
-------------------------------------------
Answer: 3
Enter author Name of Book: William Shakespeare
Enter Book Name: Hamlet
Enter new information about the book
Enter author full name: William Shakespeare
Enter title of book: Hamlet2
Enter corridor number(0-5): 2
Enter shelf number(0-4): 4
-------------------------------------------
   Welcome Administrator Default
-------------------------------------------
```

T5

```
-------------------------------------------
   Welcome Administrator Default
-------------------------------------------
   What Do you want to do ?
-------------------------------------------
   [1] Add a book
-------------------------------------------
   [2] Delete a book
-------------------------------------------
   [3] Update a book
-------------------------------------------
   [4] Add an Admin
-------------------------------------------
   [5] Print all Data
-------------------------------------------
   [0] Main Menu.
-------------------------------------------
Answer: 4
Enter a full Name: akif kartal
Enter your ID as Number: 123
Enter a password: 456
Your Administrator has added Successfully!
-------------------------------------------
   Welcome Administrator Default
-------------------------------------------
   What Do you want to do ?
-------------------------------------------
   [1] Add a book
-------------------------------------------
```

T6

```
-------------------------------------------
   Welcome Administrator Default
-------------------------------------------
   What Do you want to do ?
-------------------------------------------
   [1] Add a book
-------------------------------------------
   [2] Delete a book
-------------------------------------------
   [3] Update a book
-------------------------------------------
   [4] Add an Admin
-------------------------------------------
   [5] Print all Data
-------------------------------------------
   [0] Main Menu.
-------------------------------------------
Answer: 5
Authors:
[1] James Joyce
    Books:
    [1] Finneganın Vahı
        Locations:
        [1] c4s1.8353
    [2] Ulysses
        Locations:
        [1] c1s1.6691
[2] Harper Lee
    Books:
    [1] To Kill a Mockingbird
        Locations:
        [1] c1s3.9933
[3] Marcel Proust
    Books:
    [1] In Search of Lost Time
        Locations:
        [1] c0s1.4523
[4] Marcel Proust
    Books:
    [1] time regained
```

File - Main

```
        Locations:
        [1] c4s3.5255
[5] Fyodor Dostoyevsky
    Books:
    [1] Karamazov Kardeşler
        Locations:
        [1] c3s3.1
    [2] Crime and Punishment
        Locations:
        [1] c0s3.6618
[6] Franz Kafka
    Books:
    [1] The Trial
        Locations:
        [1] c2s3.5605
[7] Miguel de Cervantes
    Books:
    [1] Don Quixote
        Locations:
        [1] c2s0.7906
[8] Herman Melville
    Books:
    [1] Moby Dick
        Locations:
        [1] c3s1.869
[9] F. Scott Fitzgerald
    Books:
    [1] The Great Gatsby
        Locations:
        [1] c2s5.6078
[10] William Shakespeare
    Books:
    [1] Hamlet2
        Locations:
        [1] c4s2.2609
    [2] Venus and Adonis
        Locations:
        [1] c1s2.6688
    [3] The Passionate Pilgrim
        Locations:
        [1] c3s2.643
    [4] A Lover's Complaint
        Locations:
        [1] c2s2.7974
[11] Leo Tolstoy
    Books:
    [1] War and Peace
        Locations:
        [1] c1s5.897
        [2] c1s5.3942
    [2] The Prisoner of the Caucasus
        Locations:
        [1] c3s5.2635
```

T7

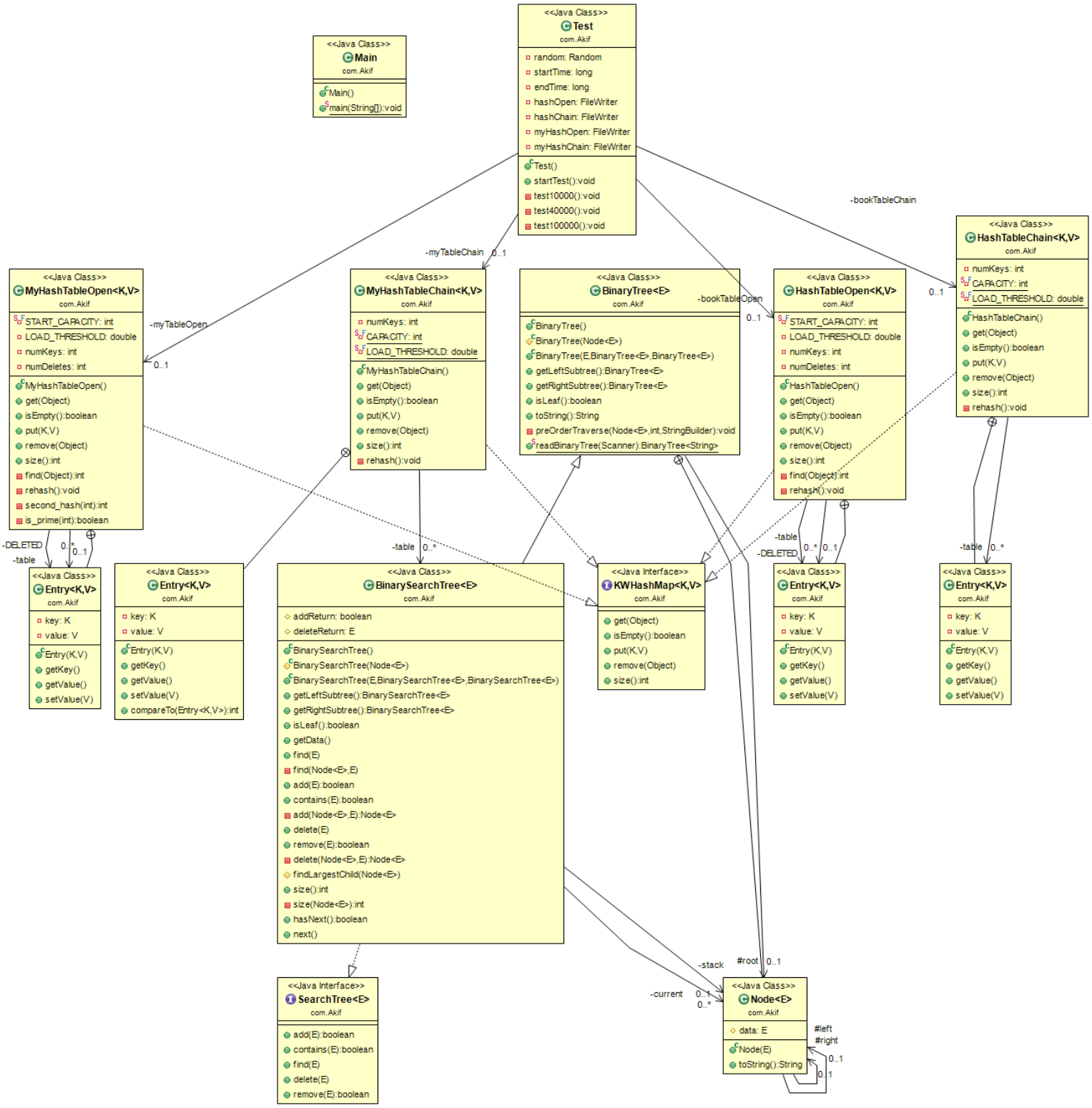
```
-------------------------------------------
   Please choose one of the option ?
-------------------------------------------
   [1] Go to Admin Panel
-------------------------------------------
   [2] Go to Search Panel
-------------------------------------------
   [0] Exit.
-------------------------------------------
Answer: 2
-------------------------------------------
   Welcome to Search Screen
-------------------------------------------
   Please choose one of the search option ?
-------------------------------------------
   [1] Search by author name
-------------------------------------------
   [2] Search by book title
-------------------------------------------
   [3] Print All Authors and Books
-------------------------------------------
   [0] Main Menu.
-------------------------------------------
Answer: 1
Enter author Name of Book: William Shakespeare
Books:
    [1] Hamlet2
    [2] Venus and Adonis
    [3] The Passionate Pilgrim
    [4] A Lover's Complaint
[0] Return Back
Choose a book to see location: 1
Locations:
    [1] c4s2.2609
-------------------------------------------
   Welcome to Search Screen
-------------------------------------------
```

T8

```
-------------------------------------------
   Welcome to Search Screen
-------------------------------------------
   Please choose one of the search option ?
-------------------------------------------
   [1] Search by author name
-------------------------------------------
   [2] Search by book title
-------------------------------------------
   [3] Print All Authors and Books
-------------------------------------------
   [0] Main Menu.
-------------------------------------------
Answer: 2
Enter Book Name: To Kill a Mockingbird
Author: Harper Lee
Locations:
    [1] c1s3.9933
Status: Available
-------------------------------------------
```

Q4 REPORT

1. CLASS DIAGRAMS



2. PROBLEM SOLUTION APPROACH

My Problem solution steps are;

– Specify the problem requirements
– Analyze the problem
– Design an algorithm and Program
– Implement the algorithm
– Test and verify the program
– Maintain and update the program

1) **Specify the problem requirements :** I understand the problem.
2) **Analyze the problem :** I identify;
   – Input data
   – Output data
   – Additional requirements and constraints

3) **Design an algorithm and Program :** I divide the problem into sub-problems. I listed major steps (sub-problems). I break down each step into a more detailed list. To do these We have to divide this big project into small pieces.

⇒ **Implement the algorithm :** I wrote the algorithm in Java by converting each step into statements of Java (classes ,methods etc.)

Firstly, I wrote the HashTableOpen and HashTableChain classes from the book. To write this class I wrote **KWHashMap** interface from the book.

Then I start to write New version of HashTableChain such that It uses chaining with **binary tree** instead of linked list. As a binary tree I wrote a complete **Binary Search Tree** Class. For this class I wrote **BinaryTree** class and **SearchTree** interface.

After, I start to write New version of HashTableOpen such that It uses an alternative probing method called **double hashing** where a second hash function is used to calculate subsequent probe locations (hash(x) + i*second_hash(x)).

4) **Test and verify the program:** To test program I wrote the **Test** Class. In this class, I fill the tables random numbers and I get 2 number from table and I save operation tine to a file. After all operations are done by using this times I draw a graph. I fill tables with size 10000, 40000 and 100000.
5) **Maintain and update the program :** I keep the program up-to-date.

3. TEST CASES

Test ID	Test Case	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
T1	Test MyHashTableOpen Class	Call the method with proper parameter	10000 random number 40000 random number 100000 random number	The table is created correct way	As Expected	Pass
T2	Test MyHashTableChain Class	Call the method with proper parameter	10000 random number 40000 random number 100000 random number	The table is created correct way	As Expected	Pass
T3	Test HashTableOpen Class	Call the method with proper parameter	10000 random number 40000 random number 100000 random number	The table is created correct way	As Expected	Pass
T4	Test HashTableChain Class	Call the method with proper parameter	10000 random number 40000 random number 100000 random number	The table is created correct way	As Expected	Pass

4. RUNNING AND RESULTS

Test ID	Test Result
T1	---------- ***My HashOpenTable with 10000 element*** 650th key of open table: 235 900th key of open table: 291 Book HashOpenTable put 10000 element and get 2 element time: 13 ms ----------
T2	---------- ***My HashChainTable with 10000 element*** 650th key of open table: 274 900th key of open table: 259 Book HashOpenTable put 10000 element and get 2 element time: 40 ms
T3	---------- ***Book HashOpenTable with 10000 element*** 650th key of open table: 775 900th key of open table: 391 Book HashOpenTable put 10000 element and get 2 element time: 18 ms ----------
T4	---------- ***Book HashChainTable with 10000 element*** 650th key of open table: 651 900th key of open table: 22 Book HashOpenTable put 10000 element and get 2 element time: 31 ms ----------