GTU Department of Computer Engineering

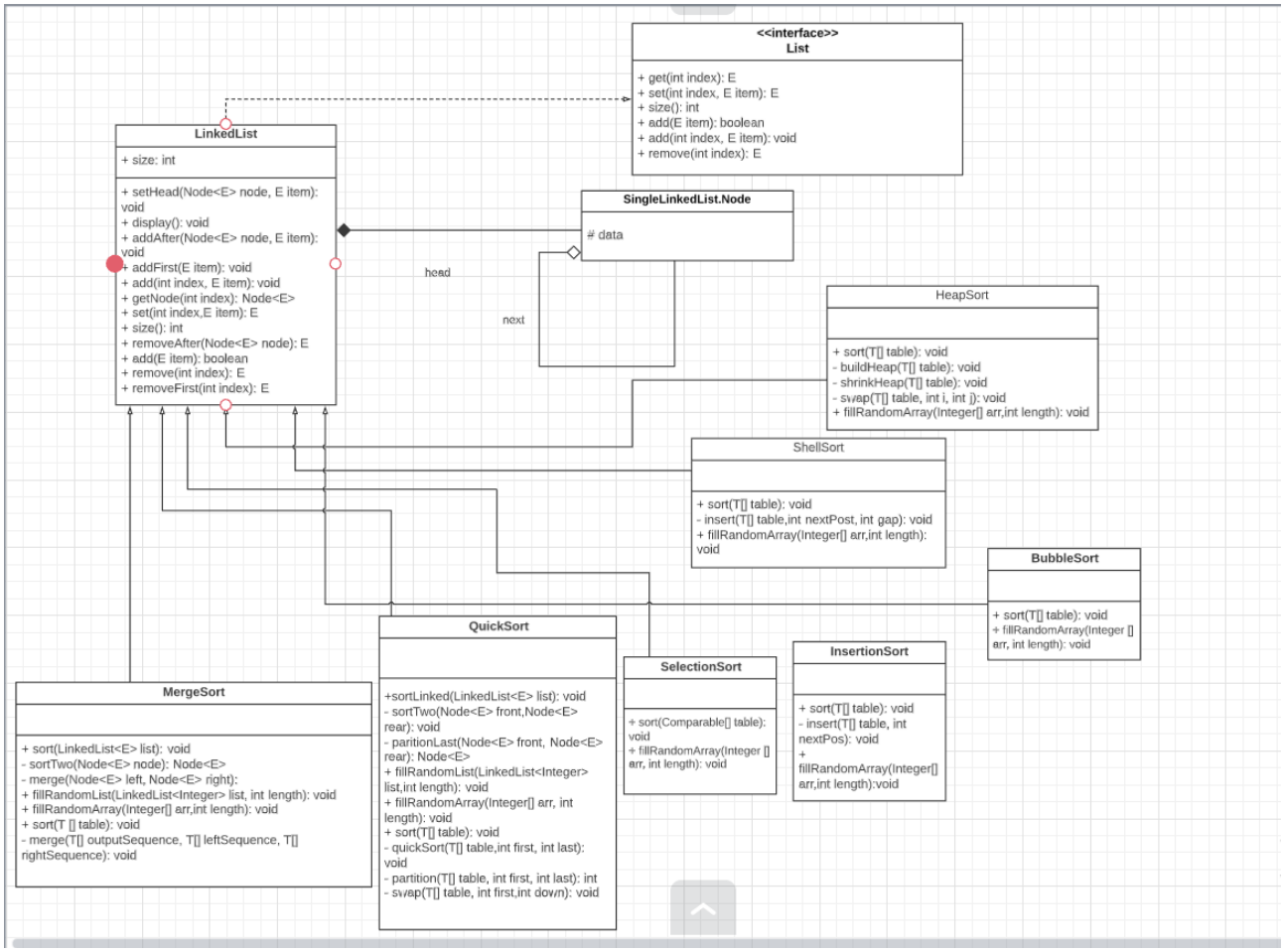
CSE 222/505 - Spring 2020

Homework 6

Mert Can BEŞİRLİ
1801042663

#Question2#

1. CLASS DIAGRAMS



2. PROBLEM SOLUTION APPROACH

In this problem, we implemented sort types as desired. Quick Sort and Merge Sort algorithms is implemented with LinkedList, used LinkedList in java. We have implemented other algorithms according to the book with arrays. Quick Sort and Merge Sort, we implemented these two as arrays again, as in the book. I created 20 random lists and 1 sorted list for each partial. I got stackOverFlowError while trying to run some lengths. This is how I compiled it-> java -Xss18m, increase the JVM memory allocation. We measured the run time for each different length. Saving run times and expected run times and compare my linked list implemention with array implemention in book.

3. TEST CASES

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
T1	Check Merge Sort List with 10000 length	Create 20 random list and 1 sorted list	61 millisecond	Showing run time	As Expected	Pass
T2	Check Merge Sort List with 40000 length	Create 20 random list and 1 sorted list	426 millisecond	Showing run time	As Expected	Pass
T3	Check Merge Sort List with 100000 length	Create 20 random list and 1 sorted list	1087 millisecond	Showing run time	As Expected	Pass
T4	Check Merge Sort List with 150000 length	Create 20 random list and 1 sorted list	1833 millisecond	Showing run time	As Expected	Pass
T5	Check Merge Sort List with 180000 length	Create 20 random list and 1 sorted list	1891 millisecond	Showing run time	As Expected	Pass
T6	Check Quick Sort List with 10000 length	Create 20 random list and 1 sorted list	2119 millisecond	Showing run time	As Expected	Pass
T7	Check Quick Sort List with 40000 length	Create 20 random list and 1 sorted list	202862 millisecond	Showing run time	As Expected	Pass
T8	Check Quick Sort	Create 20 random	1051504 millisecon	Showing run time	As Expected	Pass

	List with 100000 length	list and 1 sorted list	d			
T9	Check Quick Sort List with 150000 length	Create 20 random list and 1 sorted list	1396875 millisecond	Showing run time	As Expected	Pass
T10	Check Quick Sort List with 180000 length	Create 20 random list and 1 sorted list	2023602 millisecond	Showing run time	As Expected	Pass
T11	Check Selection Sort Array with 10000 length	Create 20 array and 1 sorted array	1436 millisecond	Showing run time	As Expected	Pass
T12	Check Selection Sort Array with 40000 length	Create 20 array and 1 sorted array	22709 millisecond	Showing run time	As Expected	Pass
T13	Check Selection Sort Array with 100000 length	Create 20 array and 1 sorted array	157531 millisecond	Showing run time	As Expected	Pass
T14	Check Selection Sort Array with 150000 length	Create 20 array and 1 sorted array	279881 millisecond	Showing run time	As Expected	Pass
T15	Check Selection Sort Array with 180000	Create 20 array and 1 sorted array	457596 millisecond	Showing run time	As Expected	Pass

	length					
T16	Check Bubble Sort Array with 10000 length	Create 20 array and 1 sorted array	2920 millisecond	Showing run time	As Expected	Pass
T17	Check Bubble Sort Array with 40000 length	Create 20 array and 1 sorted array	82720 millisecond	Showing run time	As Expected	Pass
T18	Check Bubble Sort Array with 100000 length	Create 20 array and 1 sorted array	561610 millisecond	Showing run time	As Expected	Pass
T19	Check Bubble Sort Array with 150000 length	Create 20 array and 1 sorted array	1190555 millisecond	Showing run time	As Expected	Pass
T20	Check Bubble Sort Array with 180000 length	Create 20 array and 1 sorted array	1849938 millisecond	Showing run time	As Expected	Pass
T21	Check Insertion Sort Array with 10000 length	Create 20 array and 1 sorted array	738 millisecond	Showing run time	As Expected	Pass
T22	Check Insertion Sort Array with 40000	Create 20 array and 1 sorted array	11539 millisecond	Showing run time	As Expected	Pass

	length					
T23	Check Insertion Sort Array with 100000 length	Create 20 array and 1 sorted array	63295 millisecond	Showing run time	As Expected	Pass
T24	Check Insertion Sort Array with 150000 length	Create 20 array and 1 sorted array	147361 millisecond	Showing run time	As Expected	Pass
T25	Check Insertion Sort Array with 180000 length	Create 20 array and 1 sorted array	209585 millisecond	Showing run time	As Expected	Pass
T26	Check Shell Sort Array with 10000 length	Create 20 array and 1 sorted array	46 millisecond	Showing run time	As Expected	Pass
T27	Check Shell Sort Array with 40000 length	Create 20 array and 1 sorted array	96 millisecond	Showing run time	As Expected	Pass
T28	Check Shell Sort Array with 100000 length	Create 20 array and 1 sorted array	218 millisecond	Showing run time	As Expected	Pass
T29	Check Shell Sort Array with 150000 length	Create 20 array and 1 sorted array	330 millisecond	Showing run time	As Expected	Pass
T30	Check Shell Sort Array with	Create 20 array and 1 sorted	380 millisecond	Showing run time	As Expected	Pass

	180000 length	array				
T31	Check Merge Sort Array with 10000 length	Create 20 array and 1 sorted array	167 millisecond	Showing run time	As Expected	Pass
T32	Check Merge Sort Array with 40000 length	Create 20 array and 1 sorted array	373 millisecond	Showing run time	As Expected	Pass
T33	Check Merge Sort Array with 100000 length	Create 20 array and 1 sorted array	556 millisecond	Showing run time	As Expected	Pass
T34	Check Merge Sort Array with 150000 length	Create 20 array and 1 sorted array	743 millisecond	Showing run time	As Expected	Pass
T35	Check Merge Sort Array with 180000 length	Create 20 array and 1 sorted array	768 millisecond	Showing run time	As Expected	Pass
T36	Check Heap Sort Array with 10000 length	Create 20 array and 1 sorted array	85 millisecond	Showing run time	As Expected	Pass
T37	Check Heap Sort Array with 40000 length	Create 20 array and 1 sorted array	198 millisecond	Showing run time	As Expected	Pass
T38	Check Heap Sort Array with 100000 length	Create 20 array and 1 sorted array	400 millisecond	Showing run time	As Expected	Pass

T39	Check Heap Sort Array with 150000 length	Create 20 array and 1 sorted array	563 millisecond	Showing run time	As Expected	Pass
T40	Check Heap Sort Array with 180000 length	Create 20 array and 1 sorted array	684 millisecond	Showing run time	As Expected	Pass
T41	Check Quick Sort Array with 10000 length	Create 20 array and 1 sorted array	184 millisecond	Showing run time	As Expected	Pass
T42	Check Quick Sort Array with 40000 length	Create 20 array and 1 sorted array	1213 millisecond	Showing run time	As Expected	Pass
T43	Check Quick Sort Array with 100000 length	Create 20 array and 1 sorted array	4617 millisecond	Showing run time	As Expected	Pass
T44	Check Quick Sort Array with 150000 length	Create 20 array and 1 sorted array	12145 millisecond	Showing run time	As Expected	Pass
T45	Check Quick Sort Array with 180000 length	Create 20 array and 1 sorted array	16861 millisecond	Showing run time	As Expected	Pass

4. RUNNING COMMAND AND RESULTS

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, bubble sort array with 10000 length: 2920
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, bubble sort array with 40000 length: 82720
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, bubble sort array with 100000 length: 561610
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, bubble sort array with 150000 length: 1190555
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, bubble sort array with 180000 length: 1849938
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, heap sort array with 10000 length: 85
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, heap sort array with 40000 length: 198
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, heap sort array with 100000 length: 400
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, heap sort array with 150000 length: 563
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, heap sort array with 180000 length: 684
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, insertion sort array with 40000 length: 11539
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ javac List.java Linke
ickSort.java SelectionSort.java BubbleSort.java InsertionSort.java She
st.java
Note: Some input files use unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, insertion sort array with 10000 length: 738
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, insertion sort array with 100000 length: 63295
```

```
Note: Recompile with -Xlint:unchecked for details.
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, insertion sort array with 150000 length: 147361
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, insertion sort array with 180000 length: 209585
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, merge sort array with 10000 length: 167
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ javac List.java LinkedList.java Mer
ickSort.java SelectionSort.java BubbleSort.java InsertionSort.java ShellSort.java He
st.java
Note: Some input files use unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, merge sort array with 40000 length: 373
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ javac List.java LinkedList.java MergeSort.java Qu
ickSort.java SelectionSort.java BubbleSort.java InsertionSort.java ShellSort.java HeapSort.java Te
st.java
Note: Some input files use unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, merge sort array with 150000 length: 828
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ javac List.java LinkedList.java MergeSort.java Qu
ickSort.java SelectionSort.java BubbleSort.java InsertionSort.java ShellSort.java HeapSort.java Te
st.java
Note: Some input files use unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, merge sort array with 100000 length: 556
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, merge sort array with 180000 length: 768
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, merge sort linked list with 10000 length: 61
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss4m Test

Time in milliseconds, merge sort linked list with40000 length: 426
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, merge sort linked list with 100000 length: 1087
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, merge sort linked list with 150000 length: 1833
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, merge sort linked list with 180000 length: 1891
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, quick sort array with 100000 length: 4617
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ javac List.java LinkedList.java MergeSort.java Qu
ickSort.java SelectionSort.java BubbleSort.java InsertionSort.java ShellSort.java HeapSort.java Te
st.java
Note: Some input files use unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, quick sort array with 40000 length: 1213
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ javac List.java LinkedList.java MergeSort.java Qu
ickSort.java SelectionSort.java BubbleSort.java InsertionSort.java ShellSort.java HeapSort.java Te
st.java
Note: Some input files use unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, quick sort array with 10000 length: 184
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, quick sort array with 150000 length: 12145
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, quick sort array with 180000 length: 16861
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, quick sort linked list with 10000 length: 2119
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, quick sort linked list with 40000 length: 202862
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, quick sort linked list with 100000 length: 1051504
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, quick sort linked list with 150000 length: 1396875
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, quick sort linked list with 180000 length: 2023602
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, selection sort array with 10000 length: 1436
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, selection sort array with 40000 length: 22709
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, selection sort array with 100000 length: 157531
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, selection sort array with 150000 length: 279881
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, selection sort array with 180000 length: 457596
```

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/LastSorting$ java -Xss18m Test

Time in milliseconds, shell sort array with 10000 length: 44
```

```
ubuntu@ubuntu-VirtualBox:~/MasaüstÜ/LastSorting$ java -Xss18m Test

Time in milliseconds, shell sort array with 40000 length: 96
```

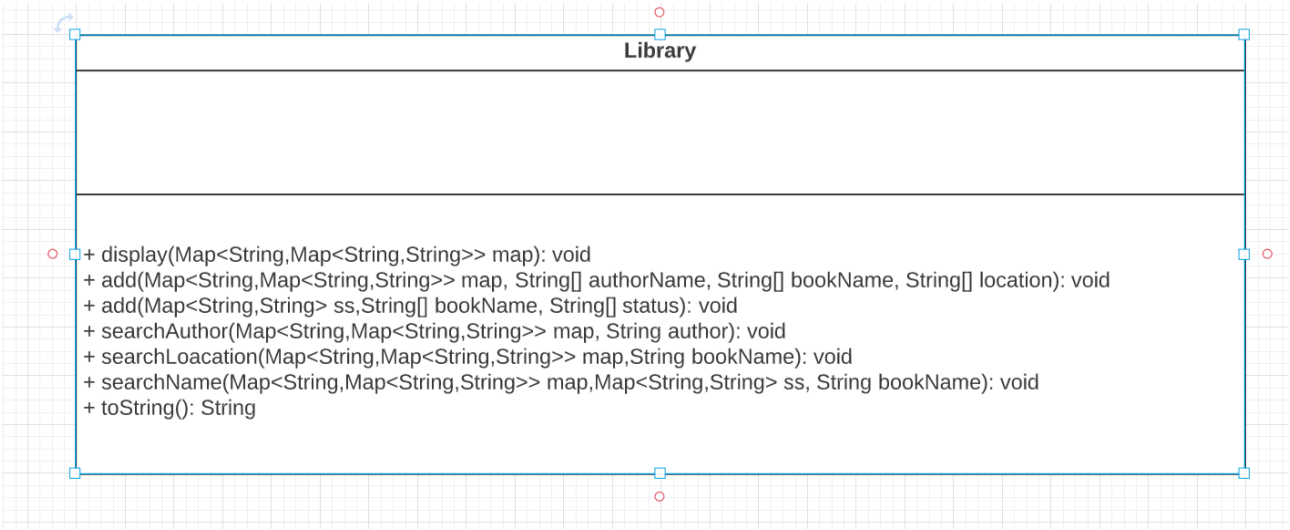
```
ubuntu@ubuntu-VirtualBox:~/MasaüstÜ/LastSorting$ java -Xss18m Test

Time in milliseconds, shell sort array with 100000 length: 218
```

```
ubuntu@ubuntu-VirtualBox:~/MasaüstÜ/LastSorting$ java -Xss18m Test

Time in milliseconds, shell sort array with 150000 length: 330
```

```
ubuntu@ubuntu-VirtualBox:~/MasaüstÜ/LastSorting$ java -Xss18m Test

Time in milliseconds, shell sort array with 180000 length: 380
```

#Question3#

1. CLASS DIAGRAMS



2. PROBLEM SOLUTION APPROACH

In this problem, we did library automation system. In my code, I determined the admins, users, author names, the names of the books, their locations. So that there is no difficulty in getting it from the user. Then they added the map. It has three users and five administrator in my code. In order to go to the menu, administrator must first be selected. Then the administrator must enter the correct password(Password: cse222). Menu is includes :Searching by author name,If admin want to adding, If admin want to removing, Searching by book title, If admin wants to update the book, Display library information, Exit. When searching by author name, 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.

3. TEST CASES

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
T1	Choose admin and enter valid password	Enter password of library system	Password is cse222	Show menu option	As Expected	Pass
T2	Choosing Searching by author name	Enter user’s name to take process then author name then book name	User1 Tolstoy Sanat Nedir?	Show choosing book’s location	As Expected	Pass
T3	Choosing Searching by book title	Enter book name	1984	Show book’s location and status	As Expected	Pass
T4	Choose If admin want to adding	Enter author name,book name and location	Franz Kafka Donusum c3s4.1234	Adding book in library	As Expected	Pass
T5	Choose If admin want to removing	Enter author name, book name	Franz Kafka Donusum	Removing book in library	As Expected	Pass
T6	Choose If admin wants to update the book	Enter author name and book name then update	Tolstoy Sanat Nedir?	Location is updated	As Expected	Pass

		location	c2s5.2222			

4. RUNNING COMMAND AND RESULTS

```
ubuntu@ubuntu-VirtualBox:~/Masaüstü/Set-Map/Hmw6$ javac Library.java Test.java
ubuntu@ubuntu-VirtualBox:~/Masaüstü/Set-Map/Hmw6$ java Test
Author name: Tolstoy
 Book name: Sanat Nedir?
 Location: c2s6.1323
 Book name: Anna Karenina
 Location: c1s2.1233

Author name: Dostoyevski
 Book name: Suc ve Ceza
 Location: c1s1.1111

Author name: George Orwell
 Book name: 1984
 Location: c2s5.1423

Author name: Cemal Sureya
 Book name: Sevda Sözleri
 Location: c3s4.1313

Author name: Jared Diamond
 Book name: Tufek Mikrop ve Celik
 Location: c1s3.1345

Which administrator wants to take action?
[Admin3, Admin2, Admin1]
Admin3
Enter password for enter library system:
cse222

Enter of choice:
S->Searching by author name
A->If admin want to adding
B->If admin want to removing
R->Searching by book title
E->If admin wants to update the book
P->Display library information
Q->Exit
```

```
Enter of choice:
S->Searching by author name
A->If admin want to adding
B->If admin want to removing
R->Searching by book title
E->If admin wants to update the book
P->Display library information
Q->Exit
S
Which user wants to take process?
[User2, User1, User5, User4, User3]
User1
Enter of author name for search:
Tolstoy
Book name: Sanat Nedir?
Book name: Anna Karenina

The user should choose which book he/she wants to see:
Sanat Nedir?
Location: c2s6.1323
```

```
Enter of choice:
S->Searching by author name
A->If admin want to adding
B->If admin want to removing
R->Searching by book title
E->If admin wants to update the book
P->Display library information
Q->Exit
R
Enter of book's name for search:
1984
Author name: George Orwell
 Location: c2s5.1423
 Status: library
```

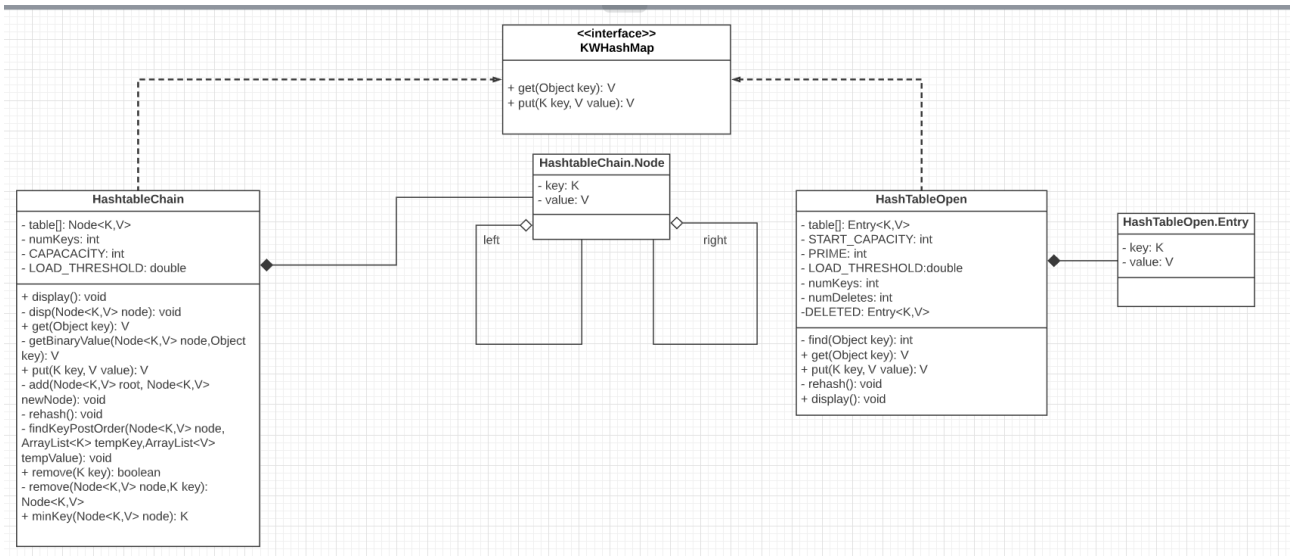
```
Enter of choice:
S->Searching by author name
A->If admin want to adding
B->If admin want to removing
R->Searching by book title
E->If admin wants to update the book
P->Display library information
Q->Exit
A
Enter author name to book:
Franz Kafka
Enter book names and location:
Donusum
c3s4.1234
```

```
Enter of choice:
S->Searching by author name
A->If admin want to adding
B->If admin want to removing
R->Searching by book title
E->If admin wants to update the book
P->Display library information
Q->Exit
B
Enter removing author name:
Franz Kafka
Enter the name of the author's book:
Donusum

Enter of choice:
```

```
P->Display library information
Q->Exit
E
Enter of author's name for update information:
Tolstoy
Which book is update, please choose:
Sanat Nedir?
Please enter of update location(for example: c2s3.1123 etc.):
c2s5.2222

Enter of choice:
S->Searching by author name
A->If admin want to adding
B->If admin want to removing
R->Searching by book title
E->If admin wants to update the book
P->Display library information
Q->Exit
P
Author name: Tolstoy
 Book name: Sanat Nedir?
 Location: c2s5.2222
 Book name: Anna Karenina
 Location: c1s2.1233

Author name: Dostoyevski
 Book name: Suc ve Ceza
 Location: c1s1.1111

Author name: George Orwell
 Book name: 1984
 Location: c2s5.1423

Author name: Cemal Sureya
 Book name: Sevda Sözleri
 Location: c3s4.1313

Author name: Jared Diamond
 Book name: Tufek Mikrop ve Celik
 Location: c1s3.1345
```

#Question4#

1.CLASS DIAGRAMS



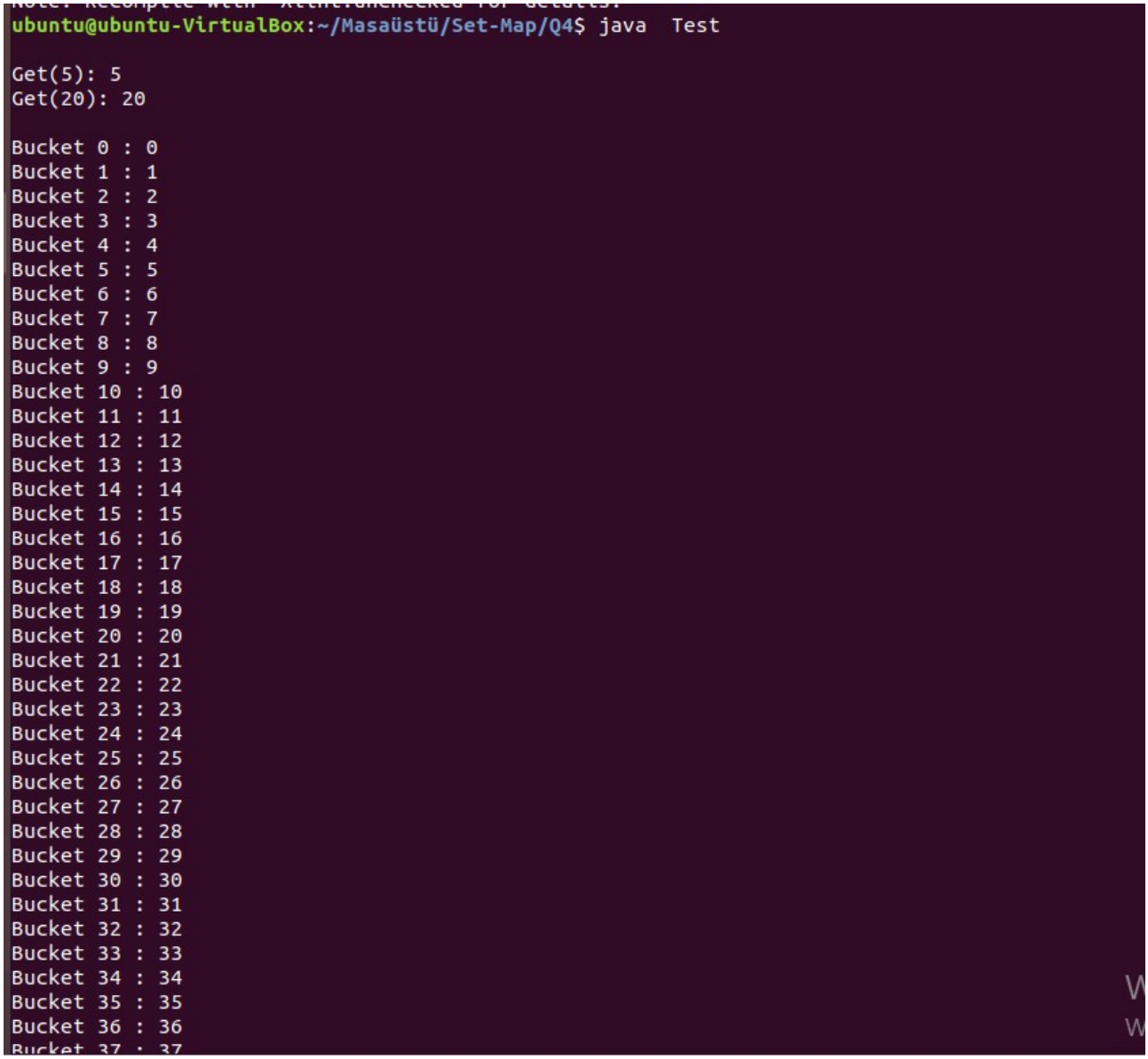
2.PROBLEM SOLUTION APPROACH

In this problem, we tried to do hashtable operations in two ways.1:Open adressing, 2: Chaining. In open addressing, if the table entry is full, subsequent probe locations are calculated using linear probing (hash(x) +i) or quadratic probing (hash(x) + i^2) methods. Use 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)). In HashTableOpen class, I wrote them in the Find method to affect them separately(linear probe, quadratic probe,double hashing) and kept them in the comment line, this can be understood.
In HashtableChaining class, using binary tree and realize operations.

3.TEST CASES

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
T1	Check open adressing and display table	Adding value to table	Key and value numbers	Show display table	As Expected	Pass
T2	Check chaining and display table	Adding value to table	Key and value numbers	Show display table	As Expected	Pass
T3	Check get key for open adressing	Find key	Key number	Show value	As Expected	Pass
T4	Check get key for chaining	Find key	Key number	Show value	As Expected	Pass

4. RUNNING COMMAND AND RESULTS



```
Bucket 38 : 38
Bucket 39 : 39
Bucket 40 : 40
Bucket 41 : 41
Bucket 42 : 42
Bucket 43 : 43
Bucket 44 : 44
Bucket 45 : 45
Bucket 46 : 46
Bucket 47 : 47
Bucket 48 : 48
Bucket 49 : 49
Bucket 50 :
Bucket 51 :
Bucket 52 :
Bucket 53 :
Bucket 54 :
Bucket 55 :
Bucket 56 :
Bucket 57 :
Bucket 58 :
Bucket 59 :
Bucket 60 :
Bucket 61 :
Bucket 62 :
Bucket 63 :
Bucket 64 :
Bucket 65 :
Bucket 66 :
Bucket 67 :
Bucket 68 :
Bucket 69 :
Bucket 70 :
Bucket 71 :
Bucket 72 :
Bucket 73 :
Bucket 74 :
Bucket 75 :
Bucket 76 :
Bucket 77 :
Bucket 78 :
Bucket 79 :
Bucket 80 :
Bucket 81 :
Bucket 82 :
```

```
Bucket 83 :
Bucket 84 :
Bucket 85 :
Bucket 86 :
Bucket 87 :
Bucket 88 :
Bucket 89 :
Bucket 90 :
Bucket 91 :
Bucket 92 :
Bucket 93 :
Bucket 94 :
Bucket 95 :
Bucket 96 :
Bucket 97 :
Bucket 98 :
Bucket 99 :
Bucket 100 :
Get(5): 5
Get(20): 20

Table: 0 : 0 0
Table: 1 : 1 1
Table: 2 : 2 2
Table: 3 : 3 3
Table: 4 : 4 4
Table: 5 : 5 5
Table: 6 : 6 6
Table: 7 : 7 7
Table: 8 : 8 8
Table: 9 : 9 9
Table: 10 : 10 10
Table: 11 : 11 11
Table: 12 : 12 12
Table: 13 : 13 13
Table: 14 : 14 14
Table: 15 : 15 15
Table: 16 : 16 16
Table: 17 : 17 17
Table: 18 : 18 18
Table: 19 : 19 19
Table: 20 : 20 20
Table: 21 : 21 21
Table: 22 : 22 22
```

```
Table: 23 : 23 23
Table: 24 : 24 24
Table: 25 : 25 25
Table: 26 : 26 26
Table: 27 : 27 27
Table: 28 : 28 28
Table: 29 : 29 29
Table: 30 : 30 30
Table: 31 : 31 31
Table: 32 : 32 32
Table: 33 : 33 33
Table: 34 : 34 34
Table: 35 : 35 35
Table: 36 : 36 36
Table: 37 : 37 37
Table: 38 : 38 38
Table: 39 : 39 39
Table: 40 : 40 40
Table: 41 : 41 41
Table: 42 : 42 42
Table: 43 : 43 43
Table: 44 : 44 44
Table: 45 : 45 45
Table: 46 : 46 46
Table: 47 : 47 47
Table: 48 : 48 48
Table: 49 : 49 49
Table: 50 :
Table: 51 :
Table: 52 :
Table: 53 :
Table: 54 :
Table: 55 :
Table: 56 :
Table: 57 :
Table: 58 :
Table: 59 :
Table: 60 :
Table: 61 :
Table: 62 :
Table: 63 :
Table: 64 :
Table: 65 :
Table: 66 :
Table: 67 :
```

```
Table: 68 :
Table: 69 :
Table: 70 :
Table: 71 :
Table: 72 :
Table: 73 :
Table: 74 :
Table: 75 :
Table: 76 :
Table: 77 :
Table: 78 :
Table: 79 :
Table: 80 :
Table: 81 :
Table: 82 :
Table: 83 :
Table: 84 :
Table: 85 :
Table: 86 :
Table: 87 :
Table: 88 :
Table: 89 :
Table: 90 :
Table: 91 :
Table: 92 :
Table: 93 :
Table: 94 :
Table: 95 :
Table: 96 :
Table: 97 :
Table: 98 :
Table: 99 :
Table: 100 :
ubuntu@ubuntu-VirtualBox:~/Masaüstü/Set-Map/Q4$
```