

Gebze Technical University  
Computer Engineering

CSE 222  
2017 Spring

HOMEWORK VII(7) REPORT

OSMAN AKKUS  
151044055

**Q1)**

### **Problem Solution Approach**

Soruyu kendi anladığım şekilde yapmaya çalıştım. Soruda bizden Navigable Map i BinarySearch Tree ile implement etmemiz istendi. Bunun için öncelikle Binary Search Tree yi ve bunun superclassı olan Binary Tree yi  $\langle K, V \rangle$  şeklinde key i ve value yu tutabilecek şekilde elverişli hale getirdim. Daha sonra put methodu ile key ve valuelar ı binary Search Tree ye add methodu ile build ettim.

Daha sonra hocanın yazdığı Q1Test fonksiyonunda yazdığım fonksiyonları tek tek test ettim.

**Q2)**

### **Problem Solution Approach**

Bu soruda anladığım HashTableChain i LinkedList objesi yerine open addressing deki gibi Entry arrayi kullandım. Entry array i aynı şekilde hashCode a göre index leri buluyor. Eğer aynı index deyse Entry class ının içinde tanımladığım Entry $\langle K, V \rangle$  next e aynı key in farklı value sunu ekliyor. Ödevi bu şekilde anladığım için bu şekilde yaptım.

Daha sonra hocanın yazdığı Q2Test fonksiyonunda bütün fonksiyonları test ettim.

### **Test Cases for both**

Test case olarak hepsinin yazılan bütün methodları Main class ındaki her iki soru için yapılan test fonksiyonlarında test edildi.

**Q1)**

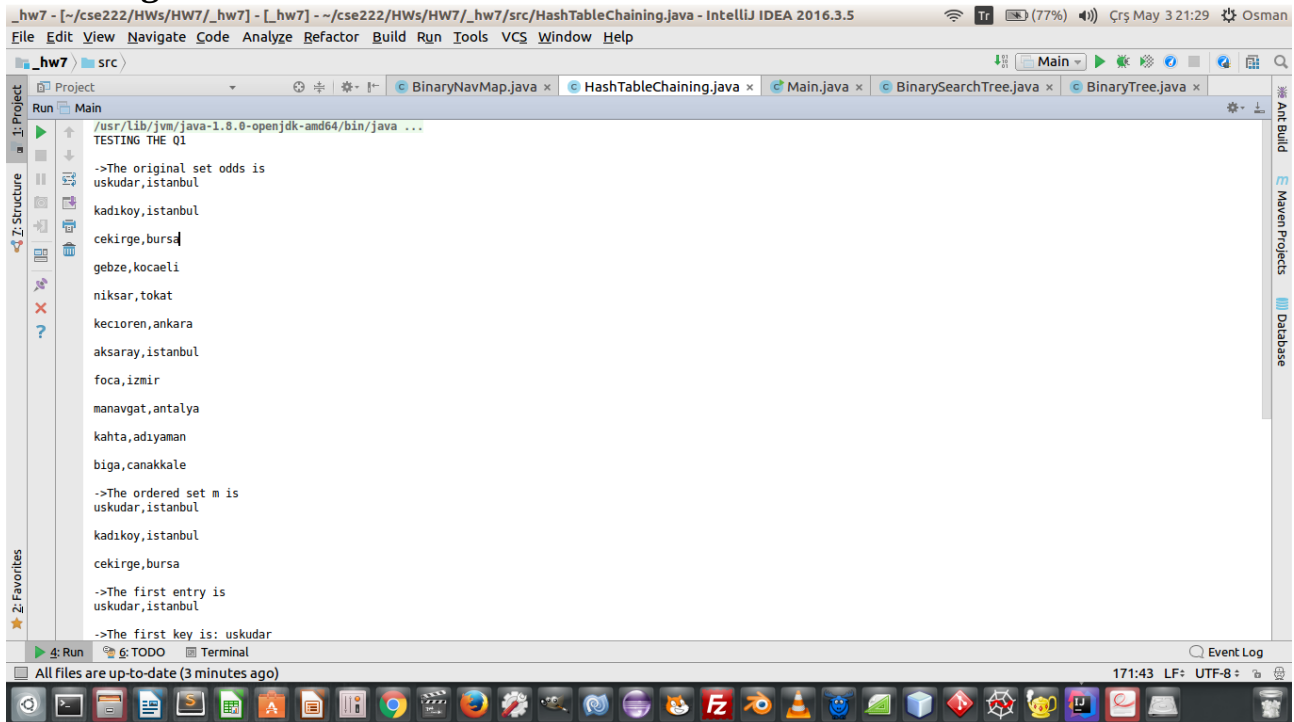
birinci soruda ilk önce bütün elemanlar put methodu ile ekleniyor  
Tüm elemanlar print ediliyor.  
Daha sonra subMap print ediliyor.

.....

**Q2)**

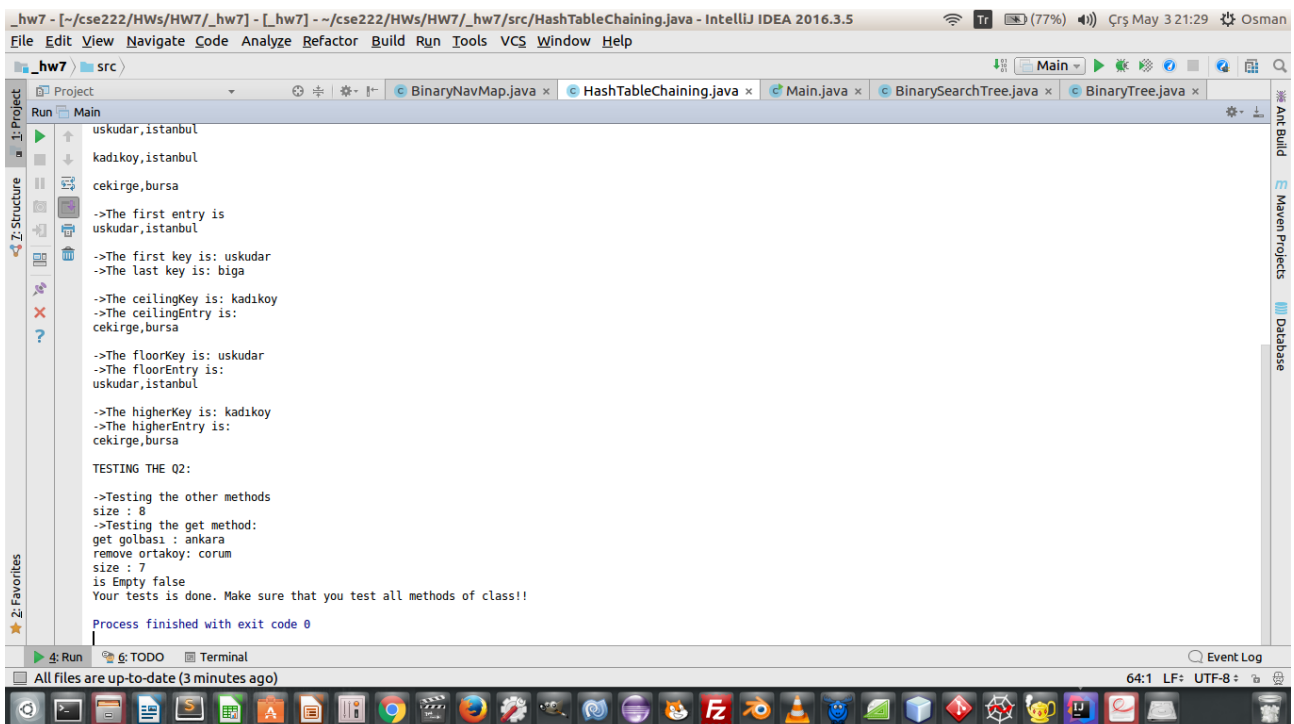
size a bakılıyor  
get methodu test ediliyor.  
Remove methodu test ediliyo  
remove dan sonra tekrar size kontrol ediliyor.  
Is Empty == false

(It is generated for both because both of them in a same package and same main)  
**Running Results for both**



```
hw7 - [~/cse222/HWs/HW7/_hw7] - [~/cse222/HWs/HW7/_hw7/src/HashTableChaining.java - IntelliJ IDEA 2016.3.5
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

Project: hw7
Run: Main
Main
/usr/lib/jvm/java-1.8.0-openjdk-amd64/bin/java ...
TESTING THE Q1
->The original set odds is
uskudar,istanbul
kadi koy,istanbul
cekirge,bursa
gebze,kocaeli
niksar,tokat
kecioren,ankara
aksaray,istanbul
foca,izmir
manavgat,antalya
kahta,adlyaman
biga,canakkale
->The ordered set m is
uskudar,istanbul
kadi koy,istanbul
cekirge,bursa
->The first entry is
uskudar,istanbul
->The first key is: uskudar
```



```
hw7 - [~/cse222/HWs/HW7/_hw7] - [~/cse222/HWs/HW7/_hw7/src/HashTableChaining.java - IntelliJ IDEA 2016.3.5
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

Project: hw7
Run: Main
Main
uskudar,istanbul
kadi koy,istanbul
cekirge,bursa
->The first entry is
uskudar,istanbul
->The first key is: uskudar
->The last key is: biga
->The ceilingKey is: kadi koy
->The ceilingEntry is:
cekirge,bursa
->The floorKey is: uskudar
->The floorEntry is:
uskudar,istanbul
->The higherKey is: kadi koy
->The higherEntry is:
cekirge,bursa
TESTING THE Q2:
->Testing the other methods
size : 8
->Testing the get method:
get golbasi : ankara
remove ortakoy: corum
size : 7
is Empty false
Your tests is done. Make sure that you test all methods of class!!
Process finished with exit code 0
```

(It is generated for both because both of them in a same package)

## Class Diagrams for both

BinaryNavMap	
mySearchTree	BinarySearchTree<K, V>
BinaryNavMap(ArrayList<Node<K, V>>)	
BinaryNavMap()	
put(K, V)	V
toString()	String
entrySet()	Set<Entry<K, V>>
lowerEntry(K)	Entry<K, V>
lowerKey(K)	K
floorEntry(K)	Entry<K, V>
floorKey(K)	K
ceilingEntry(K)	Entry<K, V>
ceilingKey(K)	K
higherEntry(K)	Entry<K, V>
higherKey(K)	K
firstEntry()	Entry<K, V>
lastEntry()	Entry<K, V>
pollFirstEntry()	Entry<K, V>
pollLastEntry()	Entry<K, V>
descendingMap()	NavigableMap<K, V>
navigableKeySet()	NavigableSet<K>
descendingKeySet()	NavigableSet<K>
subMap(K, boolean, K, boolean)	NavigableMap<K, V>
headMap(K, boolean)	NavigableMap<K, V>
tailMap(K, boolean)	NavigableMap<K, V>
comparator()	Comparator<? super K>
subMap(K, K)	SortedMap<K, V>
headMap(K)	SortedMap<K, V>
tailMap(K)	SortedMap<K, V>
firstKey()	K
lastKey()	K

