KAUNO TECHNOLOGIJOS UNIVERSITETAS

Duomenų struktūros

3 Labaratorinio darbo ataskaita

3 Variantas

Atliko Benas Montvydas Iff 9-11

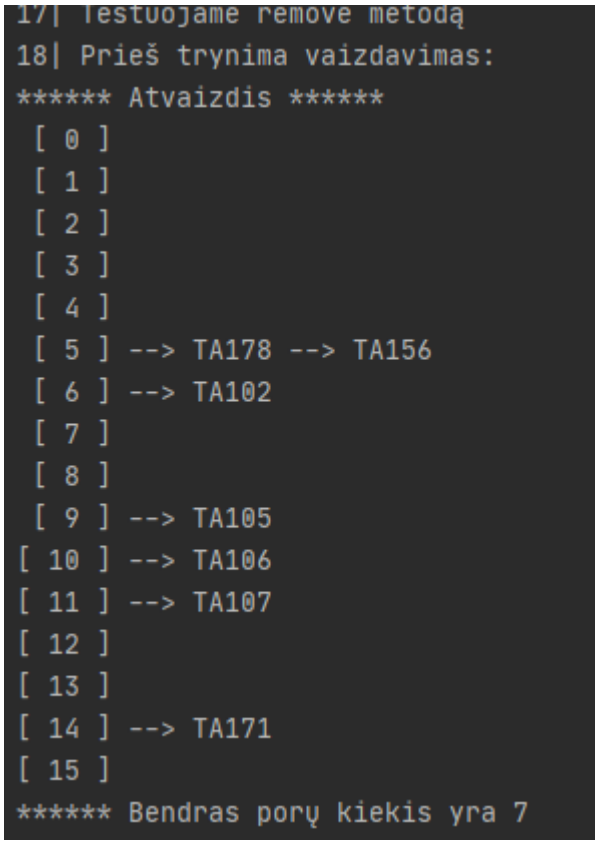
Atlikti metodai ir ištestuoti metodai:

remove(K key):

```
@Override
public V remove(K key)
{
    if (key == null) {
        throw new IllegalArgumentException("Key is null in remove(K key)");
    }
    else if(contains(key))
    {
        index = hash(key, ht); // Find the hash
        Node<K, V> prev = null;

        for (Node<K, V> node = table[index]; node != null; node = node.next) {
            if ((node.key).equals(key)) {
                if (prev == null) {
                    table[index] = node.next;
                } else {
                    prev.next = node.next;
                }
                size--;

                if (table[index] == null) {
                    chainsCounter--;
                }
                return node.value;
            }
            prev = node;
        }
    }
    return null;
}
```



```
17| Testuojame remove metodą
18| Prieš trynima vaizdavimas:
****** Atvaizdis ******
 [ 0 ]
 [ 1 ]
 [ 2 ]
 [ 3 ]
 [ 4 ]
 [ 5 ] --> TA178 --> TA156
 [ 6 ] --> TA102
 [ 7 ]
 [ 8 ]
 [ 9 ] --> TA105
[ 10 ] --> TA106
[ 11 ] --> TA107
[ 12 ]
[ 13 ]
[ 14 ] --> TA171
[ 15 ]
****** Bendras porų kiekis yra 7
```

```
19| Po trynimo:
****** Atvaizdis ******
 [ 0 ]
 [ 1 ]
 [ 2 ]
 [ 3 ]
 [ 4 ]
 [ 5 ] --> TA156
 [ 6 ] --> TA102
 [ 7 ]
 [ 8 ]
 [ 9 ] --> TA105
[ 10 ] --> TA106
[ 11 ]
[ 12 ]
[ 13 ]
[ 14 ] --> TA171
[ 15 ]
****** Bendras porų kiekis yra 5
```

containsValue(Object value):

```java
public boolean containsValue(Object value) {
    if (!(value instanceof Car))
    {
        return false;
    }
    Car tmp = (Car) value;
    for (Node<K, V> node : table)
    {
        if (node != null)
        {
            for (Node<K, V> tempNode = node; tempNode != null; tempNode = tempNode.next)
            {
                if (tmp.equals((Car)tempNode.value))
                {
                    return true;
                }
            }
        }
    }
    return false;
}
```

```
21| Patikriname ar egzistuoja containsValue
22| Kaip nariai yra:
TA156=Renault_Laguna:1997 50000 1700.0
TA102=Renault_Megane:2001 20000 3500.0
TA105=Peugeot_Partner:2008 68011 80805.6
TA106=Honda_Civic:2007 36400 8500.3
TA171=Renault_Laguna:2001 115900 7500.0

23| True testas, kai ieškome Renault_Laguna:2001 115900 7500.0
24| Atsakymas: true
25| False testas, kai ieškome Toyota_Corolla:2001 20000 8500.8
26| Atsakymas: false
```

putIfAbsent(K key, V value):

```
public V putIfAbsent(K key, V value) {
    if(!contains(key))
    {
        put(key, value);
        return null;
    }
    return value;
}
```

int numberOfEmpties:

```
public int numberOfEmpties() {
    int count = 0;
    for (Node<K, V> node : table)
    {
        if (node == null)
            count++;
    }
    return count;
}
```

```
28| Testuojame numberOfEmpties
29| Atvaizdas atrodo šitaip:
****** Atvaizdis ******
 [ 0 ]
 [ 1 ]
 [ 2 ]
 [ 3 ]
 [ 4 ]
 [ 5 ] --> TA156
 [ 6 ] --> TA102
 [ 7 ]
 [ 8 ]
 [ 9 ] --> TA105
[ 10 ] --> TA106
[ 11 ]
[ 12 ]
[ 13 ]
[ 14 ] --> TA171
[ 15 ]
****** Bendras porų kiekis yra 5
30| Number of empties 11
```

putAll(Map<K,V> map):

```
public void putAll(Map<K, V> map) {
    Set<K> mapSetKeys = map.keySet();
    for (K key: mapSetKeys) {
        V value = map.get(key);
        putIfAbsent(key, value);
    }
}
```

```
32| Testuojame Put all
33| Pradinis map
****** Atvaizdis ******
 [ 0 ]
 [ 1 ]
 [ 2 ]
 [ 3 ]
 [ 4 ]
 [ 5 ] --> TA156
 [ 6 ] --> TA102
 [ 7 ]
 [ 8 ]
 [ 9 ] --> TA105
[ 10 ] --> TA106
[ 11 ]
[ 12 ]
[ 13 ]
[ 14 ] --> TA171
[ 15 ]
****** Bendras porų kiekis yra 5
```

```
34| Naujas map:
****** Atvaizdis ******
 [ 0 ]
 [ 1 ]
 [ 2 ]
 [ 3 ] --> TA110
 [ 4 ]
 [ 5 ]
 [ 6 ]
 [ 7 ]
 [ 8 ]
 [ 9 ]
[ 10 ] --> TA106
[ 11 ] --> TA107
[ 12 ] --> TA108
[ 13 ] --> TA109
[ 14 ]
[ 15 ]
****** Bendras porų kiekis yra 5
```

```
35| Po Put all
****** Atvaizdis ******
 [ 0 ]
 [ 1 ]
 [ 2 ]
 [ 3 ] --> TA110
 [ 4 ]
 [ 5 ] --> TA156
 [ 6 ] --> TA102
 [ 7 ]
 [ 8 ]
 [ 9 ] --> TA105
[ 10 ] --> TA106
[ 11 ] --> TA107
[ 12 ] --> TA108
[ 13 ] --> TA109
[ 14 ] --> TA171
[ 15 ]
****** Bendras porų kiekis yra 9
```

replace(K key, V oldValue, V newValue):

```
public boolean replace(K key, V oldValue, V newValue)
{
    if (contains(key))
    {
        index = hash(key, ht); // Find the hash
        for (Node<K, V> node = table[index]; node != null; node = node.next) {
            if ((node.key).equals(key)) {
                if (node.value.equals(oldValue))
                {
                    node.value = newValue;
                    return true;
                }
                else
                    {
                        return false;
                    }
            }
        }
    }
    return false;
}
```

```
37| Replace metodas
38| Pradinis map
TA110=Mazda_CX-3:2006 171765 3037.4
TA156=Renault_Laguna:1997 50000 1700.0
TA102=Renault_Megane:2001 20000 3500.0
TA105=Peugeot_Partner:2008 68011 80805.6
TA106=Honda_Civic:2007 36400 8500.3
TA107=Honda_Civic:2007 36400 8500.3
TA108=Renault_Laguna:2001 115900 7500.0
TA109=Renault_Twingo:1997 158654 44265.3
TA171=Renault_Laguna:2001 115900 7500.0

39| Replace TA107 Honda_Civic:2007 36400 8500.3 į Peugeot_508:2006 187157 29338.4
40| Po replace
TA110=Mazda_CX-3:2006 171765 3037.4
TA156=Renault_Laguna:1997 50000 1700.0
TA102=Renault_Megane:2001 20000 3500.0
TA105=Peugeot_Partner:2008 68011 80805.6
TA106=Honda_Civic:2007 36400 8500.3
TA107=Peugeot_508:2006 187157 29338.4
TA108=Renault_Laguna:2001 115900 7500.0
TA109=Renault_Twingo:1997 158654 44265.3
TA171=Renault_Laguna:2001 115900 7500.0
```

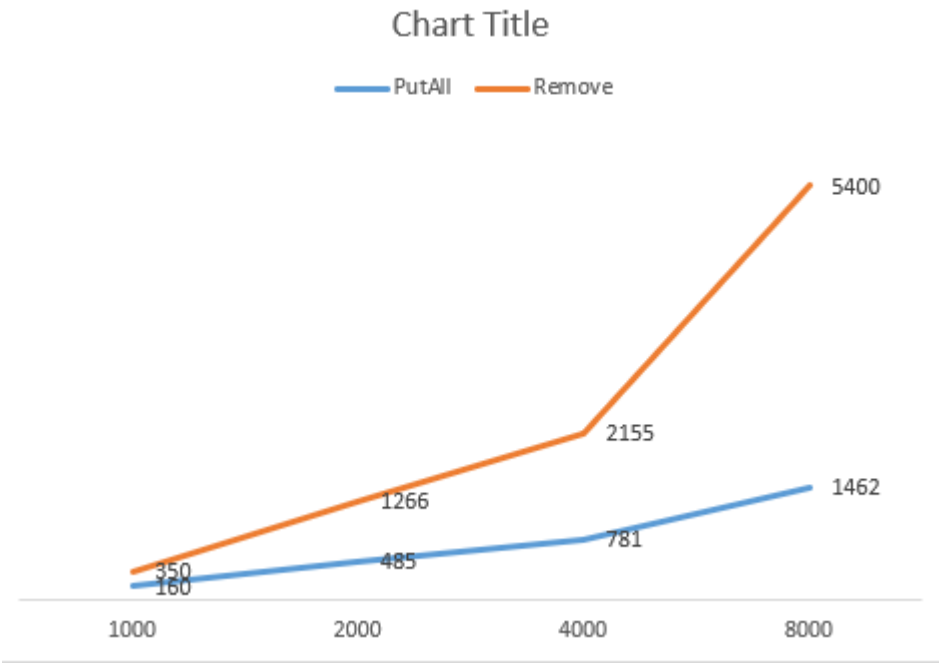
Greitaveikos testavimas:

3 variantas

3	Class HashMap: putAll()	Class HashMap: remove()

Rezultatai :

```
Benchmark          (elementCount)  Mode  Cnt     Score      Error  Units
Benchmark.putAll            10000  avgt    5   160,926 ±   43,791  us/op
Benchmark.putAll            20000  avgt    5   485,992 ±  119,543  us/op
Benchmark.putAll            40000  avgt    5   781,069 ±  188,751  us/op
Benchmark.putAll            80000  avgt    5  1462,816 ±  527,237  us/op
Benchmark.remove            10000  avgt    5   350,290 ±   51,892  us/op
Benchmark.remove            20000  avgt    5  1266,164 ±  323,343  us/op
Benchmark.remove            40000  avgt    5  2155,114 ±  848,960  us/op
Benchmark.remove            80000  avgt    5  5400,557 ± 3198,830  us/op
```



Išvados: PutAll asimptotinis sudėtingumas O(n), o remove O(n) (nes daug kolizijų, galėjo gautis) asimptotinis sudėtingumas. Gautas greitis, priklauso nuo mano kompiuterio parametro, ir ką veikiau to momentu, bet palyginimas rezultatai neturėtu keistis. Remove metodas yra lėtesnis, nei PutAll.

Kompiuterio parametrai:

Processor:	Intel(R) Core(TM) i5-8300H CPU @ 2.30GHz 2.30 GHz
Installed memory (RAM):	8,00 GB (7,88 GB usable)
System type:	64-bit Operating System, x64-based processor