TASK 4 - HashTable

Hello,

I checked your problem. Indeed, I verified that the issue exists in the current version of the HashTable implementation. This problem is easy to repair and I give you some steps, screenshot and repair project.

- 1. Change parameter
 - a. Go to HashTable.cs
 - b. Find row 162
 - c. Change parameter Capacity to newCapacity

```
р <u>-</u> в х
ELE EDIT YIEW TELERIK JUSTCODE JUSTŢRACE PROJECT BUILD DEBUG TEAM SQL JOOLS TEŞT ARCHITECTURE MINDSCAPE AMALYZE WINDOW HELP
  O - O | 簡 - 編 編 P プ・ペート Start - Debug - | 月 章 価 情 な ち 用 知 知 真 Å Performance Profiler 。
                  + X HashTableEnumerator.cs
                                                        Sample.cs
                                        - \boxed{ ^{@}_{a} \text{IncreaseCapacity()} } \\ \text{KeyValuePair<K, T> nextPair = new KeyValuePair<K, T> (key, value);} \\ \text{node.Value = nextPair;} \\
                                                                                                                                                                                                 © © 6 5 - e 0 0 0 0

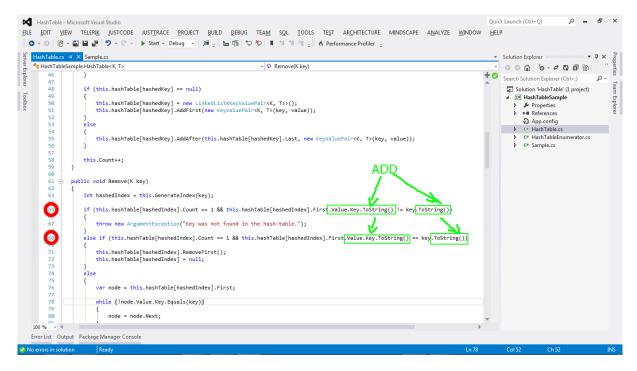
    Search Solution Explorer (Ctrl+;)

■ Search Solution Explorer (Ctrl+;)
                                                                                                                                                                                                 Solution 'HashTable' (1 project)

BashTableSample
        141
142
143
144
145
146
147
150
151
152
153
154
155
156
157
168
161
162
163
164
165
166
167
168
170
171
172
                                                                                                                                                                                                     ▶ Properties
▶ ■■ References
                          public void Clear()

    App.config
    C" HashTable.cs
    C" HashTableEnumerator.cs
    C" Sample.cs
                               this.Count = 0;
this.hashTable = new LinkedList<KeyValuePair<K, T>>[this.Capacity];
                              LinkedList<KeyValuePair<K, T>>[] copiedArray = new LinkedList<KeyValuePair<K, T>>[newCapacity];
                               for (int i = 0; i < this.hashTable.LerChange this parameter with newCapacity
                                   copiedArray[i] = this.hashTable[i];
                              ReAddValues(copiedArra) Capacity);
                          private int GenerateIndex(K kev)
                              return Math.Abs(key.GetHashCode() % this.hashTable.Length);
                           private void ReAddValues(LinkedList<KeyValuePair<K, T>>[] copiedArray, int capacity)
                               this.Capacity = capacity;
this.hashTable = new LinkedList<KeyValuePair<K, T>>[this.Capacity];
```

- 2. Add toString method
 - a. Go to HashTable.cs
 - b. Find row 65
 - c. Replace this.hashTable[hashedIndex].First with this.hashTable[hashedIndex].First.Value.Key.ToString()
 - d. Replace key with key.ToString()
 - e. Make steps C and D for row 69



I hope this information helps. I will be glad to assist you further.

Regards XXX