

Homework 7

Djuro Radusinovic

171044095

Question 2

Problem-solution approach

Here I implemented the skip list that can contain a certain number of elements. These elements are organized like the elements in a B tree. Order of the list is passed as a parameter when list is being formed. Maximum number of elements is order-1 as it is in the B tree and the minimum number of is also as it is in the B tree or $(\text{order}-1)/2$. Since it is a list, duplicate elements are allowed here.

Search method is provided that would find a node that is preceding the node in which the item is placed. The only exception is when the item is in the head node where search will return an array of nodes whose first node is the node containing our element.

When inserting first the 'preceding' list of nodes is found. After that the size is increased and maxLevel will be updated if needed. Then we will check if the element is to be added at head node. If so, if head does not contain maximum number of keys we can add it directly. Otherwise popping the last element from the head will be performed (last element in the head is larger than the element to be inserted since succeed node returned is head here). Then we continue with the regular adding which is as follows:

First case is if we are adding to the end and if the end node does not contain maximum number of element we will just add it, otherwise the new node with element to be added is created. Then popping from the last end node is performed and those elements are added to this node. After that links are updated.

Second case when it's added somewhere in between and the number of keys of that node is not maximum addition is directly performed.

Third case when there are maximum number of keys in the node it should be inserted into. In that case a new node will be made with data to be added into it. Then popping from the node that is full will be performed until new node has minimum number of keys. Afterwards some swaps between these nodes is performed so that the order stays as it is and relinkings is performed.

During removal also the search method is used.

When head is the only node left removal are done directly or when removal is in the head node and it has more than minimum number of keys. If head has minimum number of keys than the element will be swapped with the smallest element in the succeeding node. After that regular deletion will be performed which is as follows:

When node it is in has more than the minimum number of keys we can safely just remove target data from it. Otherwise if its predecessor has more than minimum number of elements, we can borrow it from it and remove the data. When both have minimum number of elements, target data will be removed and those two will be merged together.

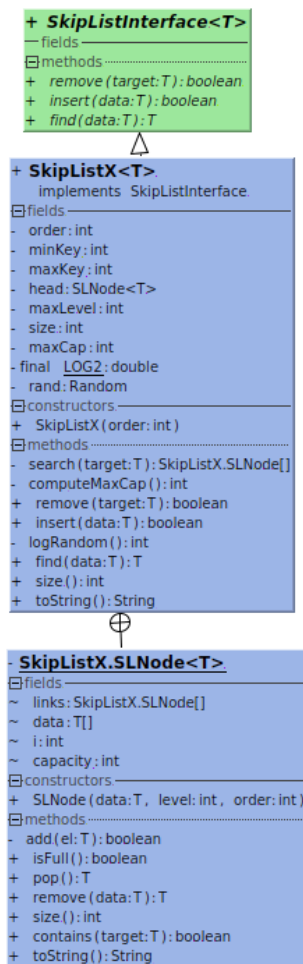
Tests cases

Test cases are run inside the virtual machine provided. Its actual results can be confirmed from the attached screenshots.

Test Scenario	Expected Results	Actual Results
Creating a skip list with order 5(meaning mini el 2 and max el 4	List should be created	As expected
Inserting 5,6,6,8,9 into the list	Should insert these 5 elements	As expected
Insert elements from 0 to 100 into the list	All of them should be inserted(covers almost all of the cases)	As expected
Now inserting 12,3,3,4 and -25	These elements are inserted in the same time covering all the remaining cases	As expected
Now removing 3	3 should be removed	As expected
Removing 3 again	3 should be removed	As expected
Removing 3 again	3 is removed successfully	As expected
Removing -25	-25 is removed	As expected
Removing 5	5 is removed	As expected
Removing 2	2 is removed	As expected
Removing 97	97 is removed	As expected
Removing 96	96 I removed	As expected
Removing 98	98 should be removed	As expected
Removing elements from 99 to 0	Should remove each element	As expected
Removing 6,8,9,12	All of them should be removed	As expected
Removing 6	6 is removed	As expected
Removing 4	4 is removed and list is now empty	As expected

Note: all the cases mentioned in the problem solution approach are tested with these cases

Class diagram of SkipListX



Running command and results

Windows taskbar showing search bar, task icons, and system tray with date/time (13:44, 2.06.2020).

Windows taskbar showing search bar, taskbar icons (File Explorer, Edge, etc.), and system tray (date, time, language).

```
Seç Komut İstemi
4 6 ----> 6 8 ----> 9 12
Finding 6
6 is found
Finding 15
15 is not in the list
Removing 6,9,12
4 6
Removing 6
4
Removing 4
Size of the list is 0
C:\Users\cse222\Desktop\HAK7\src>
```

Aramak için buraya yazın

13:44
2.06.2020