Report Assignment 1

Amelie Löwe, Håkan Johansson

October 3, 2019

1 Solution

The data structure used for the sequenceWithMinimum exercise was a double linked list. We choose this approach since the problem was to be able to execute operations from both sides of the list. We also implemented that each node keeps track of its previous and next node. When inserting new nodes, we compare the value of the new node to the value of the head- and tail node, since then we can always return the minimum value in constant time.

2 Analysis of Operations

2.1 Insert Right and Insert Left

When inserting left and right we compare the head- and tail value to the value we want to insert. Run this comparison on both of the insert methods and update the minimum value if the new value is less than the head- or tail value. By this approach we only need a few extra operations for insertion, but we will still have the worst case time complexity $O(1)$.

2.2 Remove Right and Remove Left

With the head and tail approach it is simply the same as the insert methods, but we remove head and tail instead of inserting new elements at the ends of the list. This is always time complexity $O(1)$.

2.3 Find Minimum

Worst case scenario for finding the minimum value would be to traverse the whole list and it results in time complexity $O(N)$. Since we did a little bit of more work when inserting the elements and always checked if the head- or tail value was bigger than the value to be inserted, and updated the minimum if that was the case. In this way we always keep track of the minimum and can return the value in $O(1)$.