Gebze Technical University
Computer Engineering

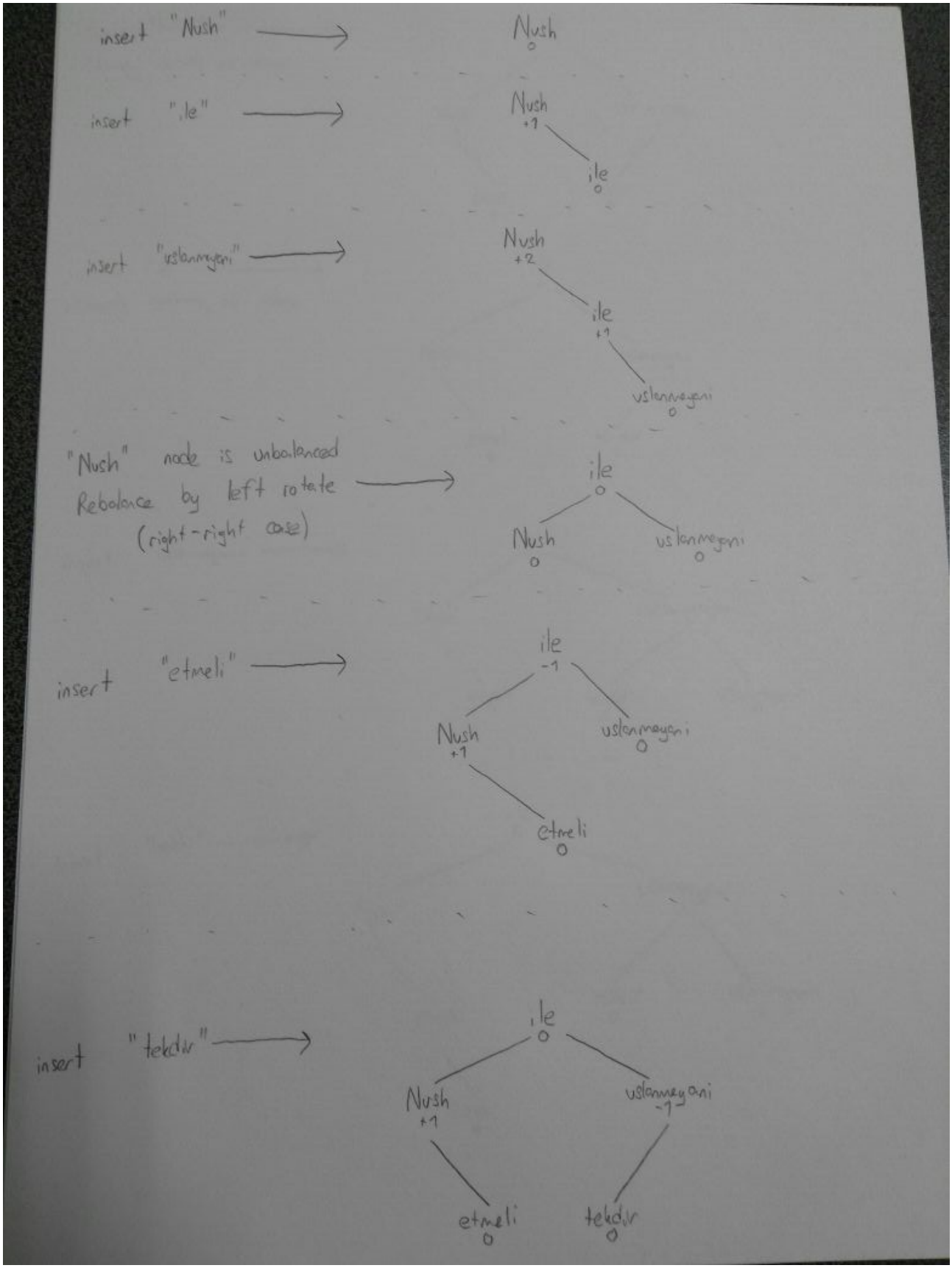
CSE 222
2017 Spring

HOMEWORK 8 REPORT

Deniz Can Erdem Yılmaz
151044001

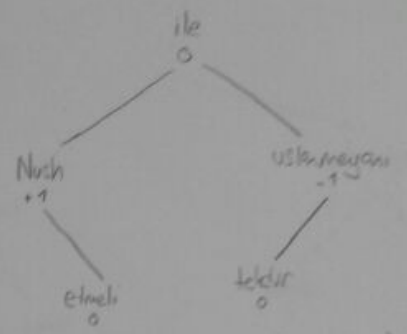
Course Assistant: Şeyma Yücer

Q1.


insert "Nush"
Nush
0
insert "ile"
Nush
+1
ile
0
insert "uslanmayan"
Nush
+2
ile
+1
uslanmayan
0
"Nush" node is unbalanced
Rebalance by left rotate
(right-right case)
ile
0
Nush
0
uslanmayan
0
insert "etmeli"
ile
-1
Nush
+1
uslanmayan
0
etmeli
0
insert "tekdir"
ile
0
Nush
+1
uslanmayan
-1
etmeli
0
tekdir
0

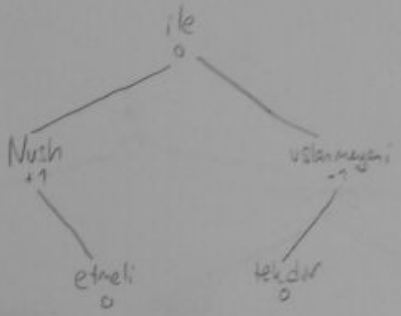
insert "tekdir" ⟶

Allready contains, no change

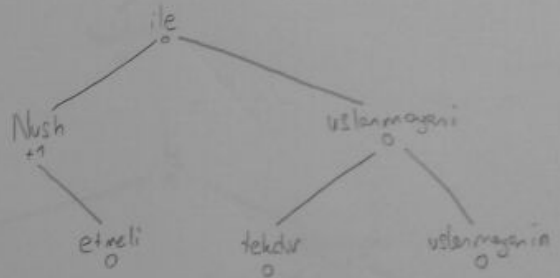


insert ".le" ⟶

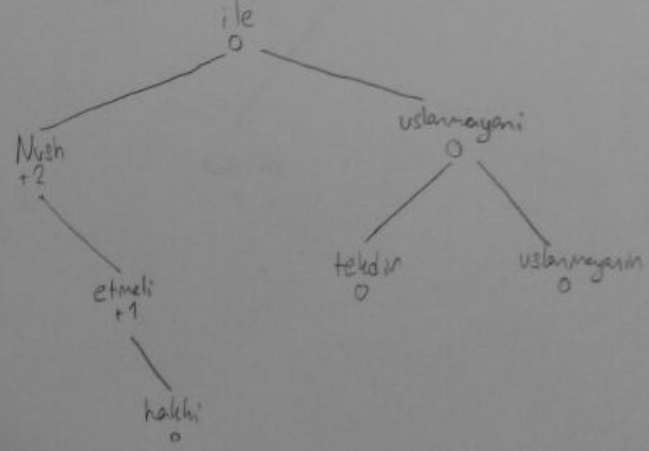
Allready contains, no change

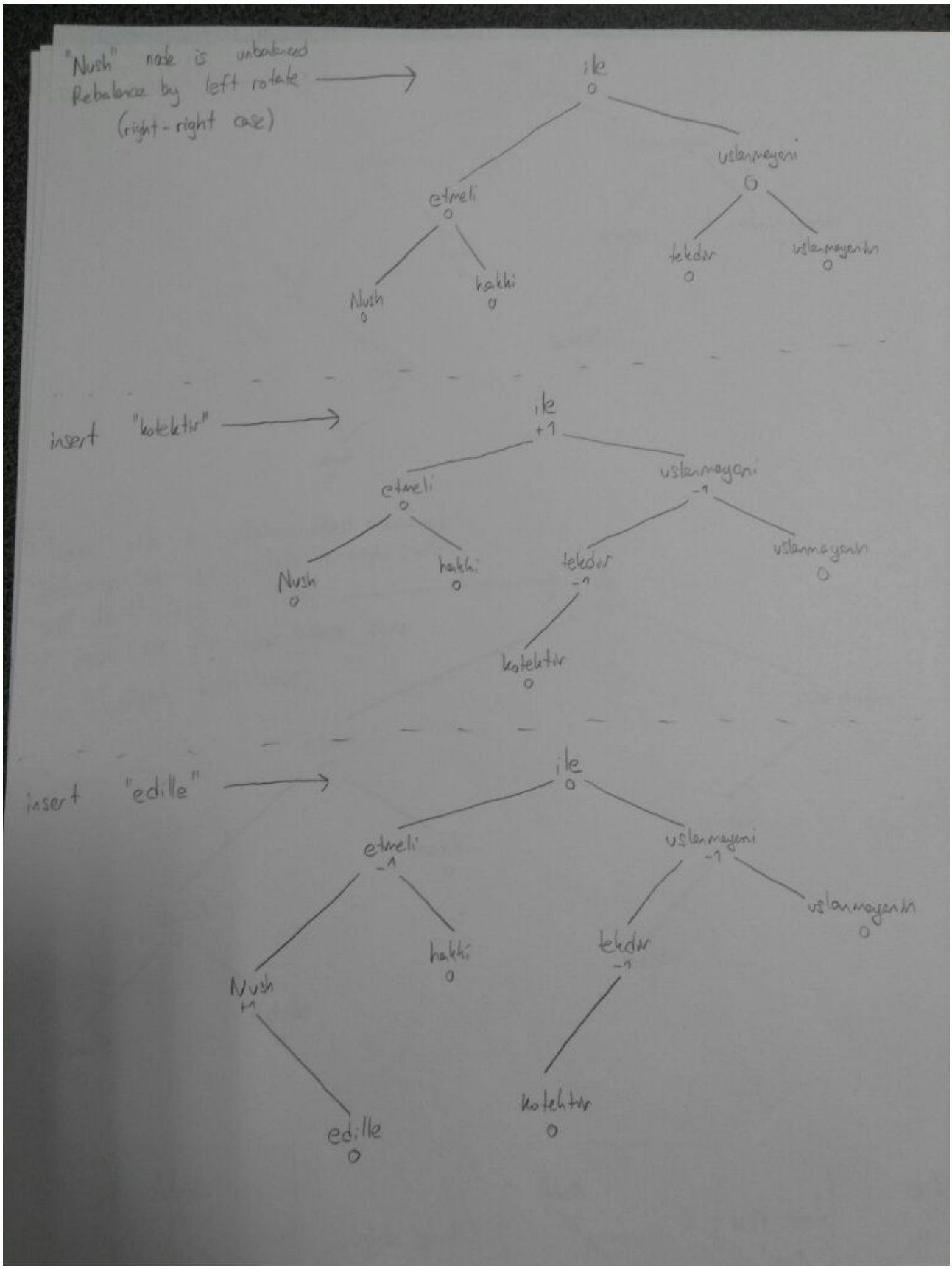


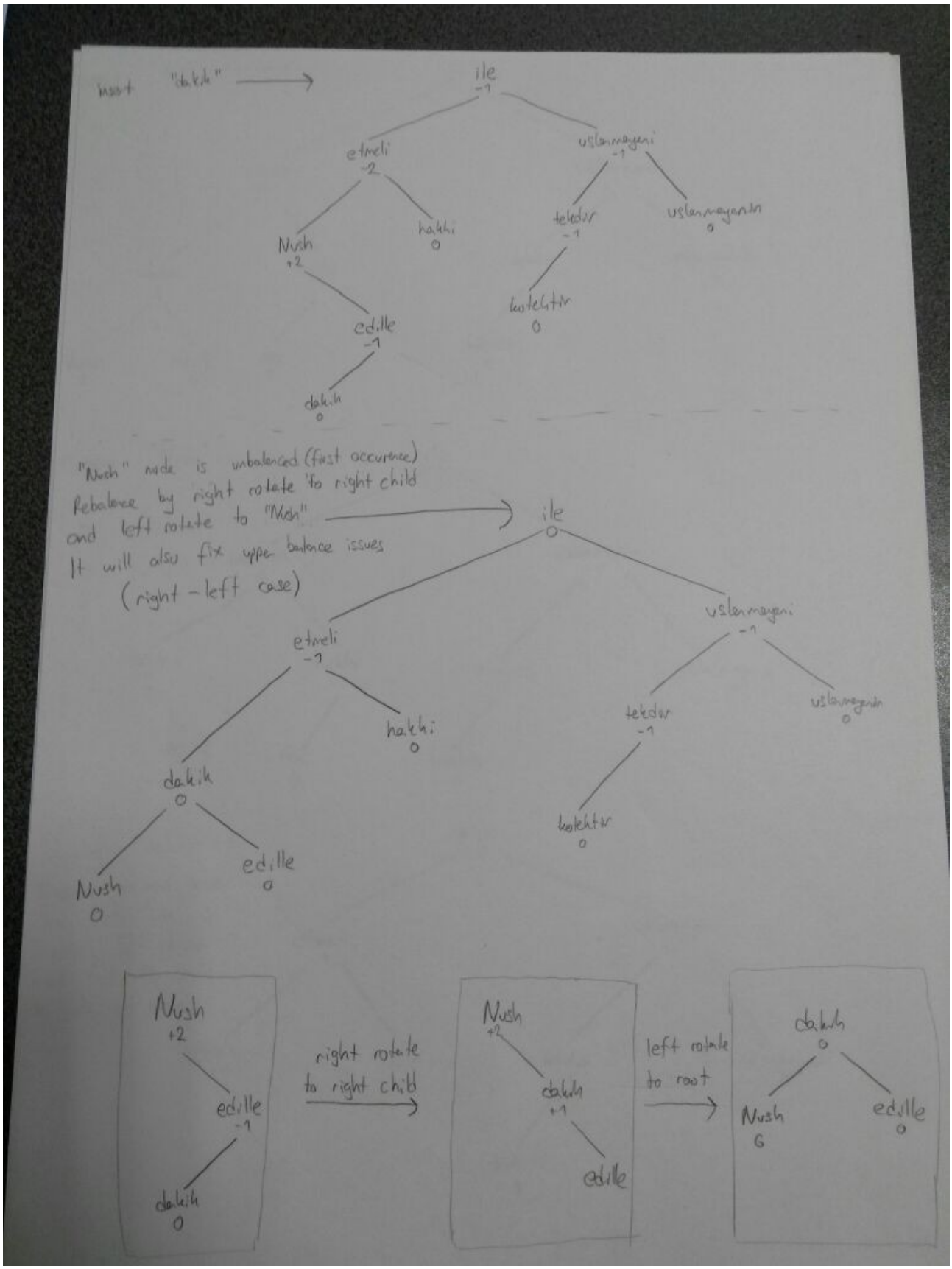
insert "uslanmayanin" ⟶

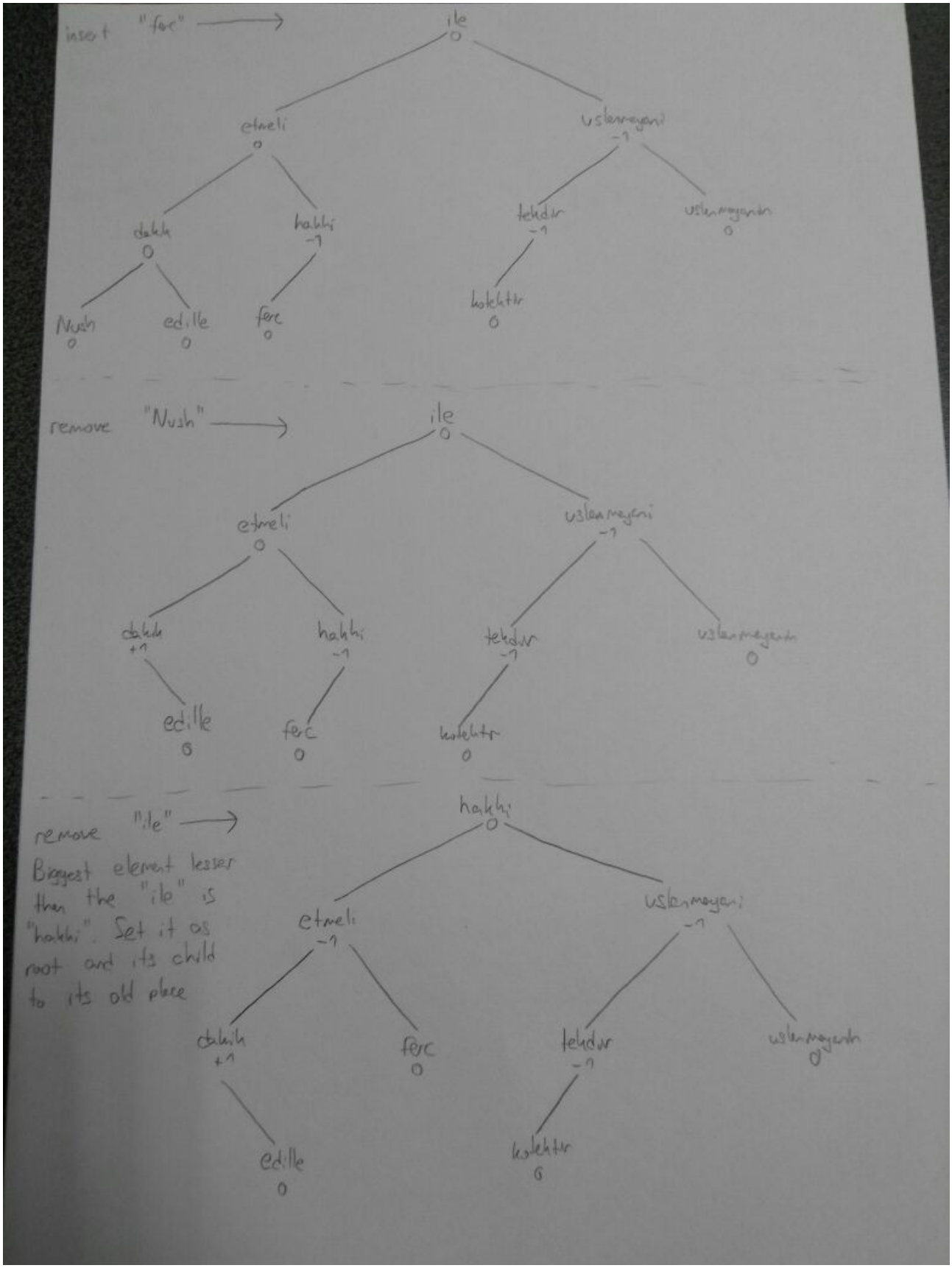


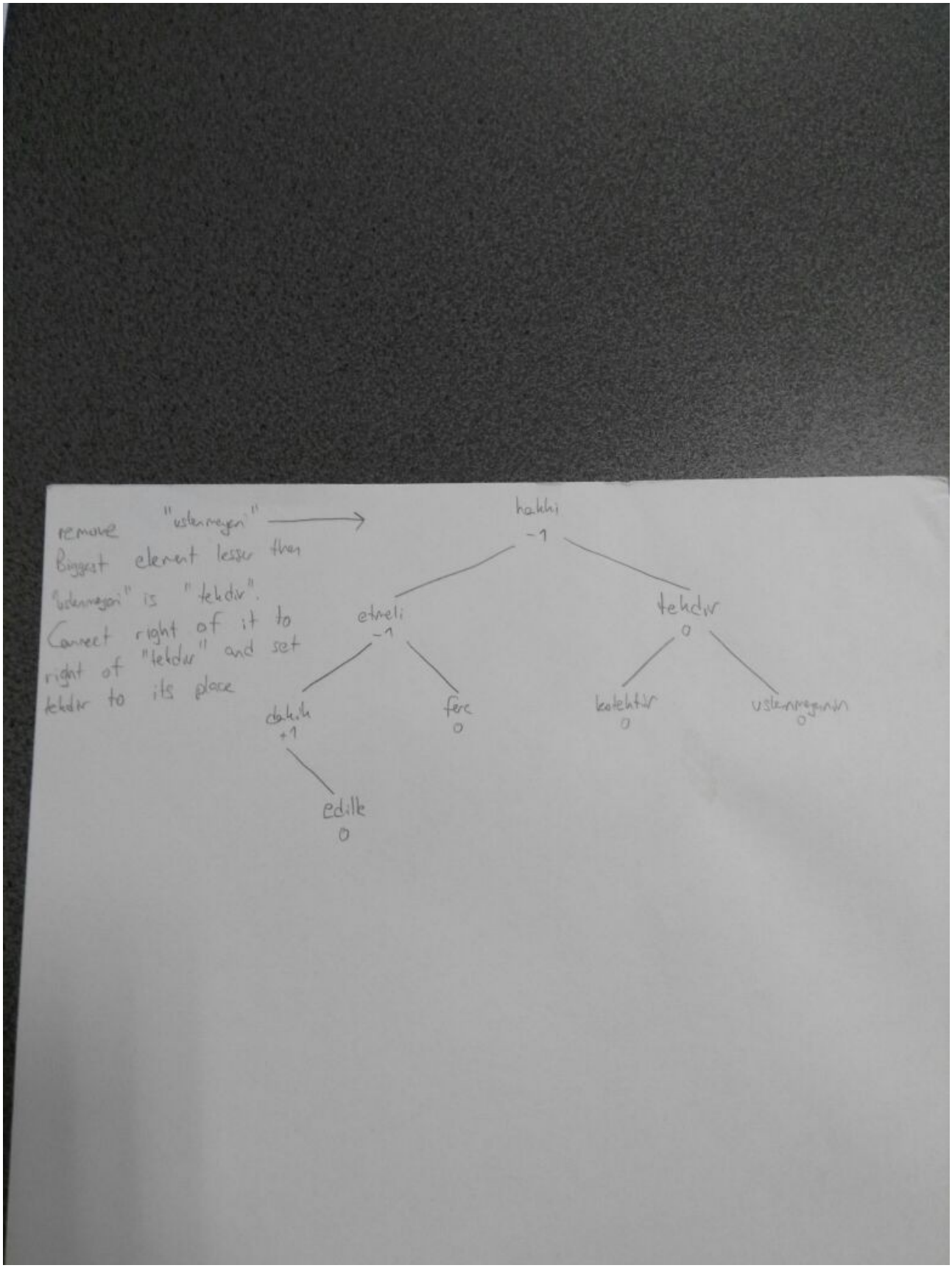
insert "hakki." ⟶



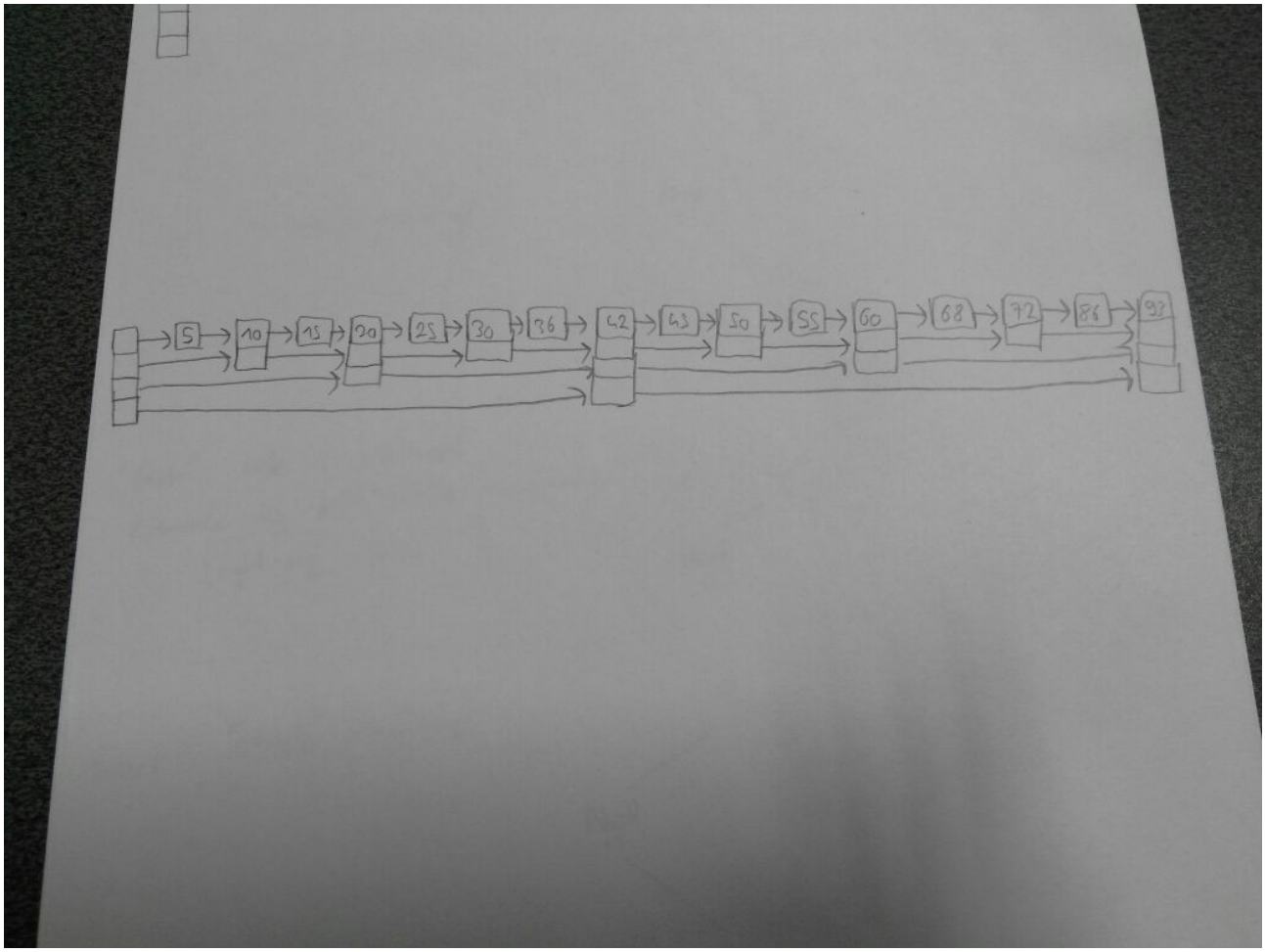

"Nush" node is unbalanced
Rebalance by left rotate
(right-right case)
ile
0
etmeli
0
uslanmayeni
0
Nush
0
hakki
0
tekdir
0
uslanmayanlr
0
insert "kotektir"
ile
+1
etmeli
0
uslanmayeni
-1
Nush
0
hakki
0
tekdir
-1
uslanmayanlr
0
kotektir
0
insert "edille"
ile
0
etmeli
-1
uslanmayeni
-1
Nush
+1
hakki
0
tekdir
-1
uslanmayanlr
0
edille
0
kotektir
0


insert "dakik"
ile
-1
etveli
-2
uslanmayan
-1
hakki
0
Nush
+2
tekdir
-1
uslanmayanlar
0
edille
-1
kotekdir
0
dakik
0
"Nush" node is unbalanced (first occurence)
Rebalance by right rotate to right child
and left rotate to "Nush"
It will also fix upper balance issues
(right - left case)
ile
0
etveli
-1
uslanmayan
-1
hakki
0
tekdir
-1
uslanmayanlar
0
dakik
0
kotekdir
0
Nush
0
edille
0
Nush
+2
edille
-1
dakik
0
right rotate
to right child
Nush
+2
dakik
+1
edille
left rotate
to root
dakik
0
Nush
0
edille
0


insert "ferc"
ile
0
etmeli
0
uslenmeyeni
-1
dahili
0
hakki
-1
tehdir
-1
uslenmeyenin
0
Nush
0
edille
0
ferc
0
kotehtr
0
remove "Nush"
ile
0
etmeli
0
uslenmeyeni
-1
dahili
+1
hakki
-1
tehdir
-1
uslenmeyenin
0
edille
0
ferc
0
kotehtr
0
remove "ile"
Biggest element lesser than the "ile" is "hakki". Set it as root and its child to its old place
hakki
0
etmeli
-1
uslenmeyeni
-1
dahili
+1
ferc
0
tehdir
-1
uslenmeyenin
0
edille
0
kotehtr
0


remove "uslenmegen" →
Biggest element lesser than "uslenmegen" is "tekdir".
Connect right of it to "tekdir" and set right of tekdir to its place
hakki
-1
etveli
-1
tekdir
0
dahith
+1
ferc
0
kotehtir
0
uslenmegenin
0
edilk
0

Q2.



5. Problem Solutions Approach

The problem of the basic binary search trees are that when a sequential data group has inserted to structure it builds something like a linked list. Which makes search linear, worst case of a binary search tree. Whenever an element has inserted or deleted checking the height level of nodes and doing necessary changes would prevent this bad situation. AVLTree is one example of that trees. Based on a rotatable binary search tree (which based on a regular binary search tree) nodes have balance values to increment for each right level and decrement for each left level. If absoulte balance value of a node is bigger than 1 a balance method has called to rebalance tree. There are 4 unbalance situations can occur: right-right, right-left, left-right, left-left. A propriate methods called if unbalance is strong with left or right, depending on the sign of balance value.

Changes:

No changes made to the incrementBalance and decrementBalance methods for removal. However a small condition check added to rotate methods to prevent a misvalue for new left/right node's balance value after rotation. Instead of changing the rebalance methods remove methods has overrided and added balance checks to prevent unbalance when an item has deleted from tree. Everything else has kept same as insertion algorithm of book.

6. Test Cases

There is no such error cases to handle new. Insertion and removal methods are implemented to handle the situtaion if a non existing element tried to remove or an already existing element has tried to inserted again. Remove method of BinarySearchTree Iterator class throws UnsupportedOperation exception and must be handled by the user if used. Giving a generic type other than the one instantiated with the constructor of AVLTree class to any public method as input won't compile in many cases and is again in the hands of user. These were the cases where program could crush. To test reasonable cases book's own examples and some additional tests has made. More information below

7. Running and Results

A sequential array, words of the sentence “The quick brown fox jumps over the lazy dog”, tree rotation example with numbers in book and another test with same numbers in random order has tested to see if program works fine. Test and results can be found under the SS folder.