**4-Binary Trees**

**Writing exercises**

1.

a)

Inorder: A B C D E F G

Preorder: D B A C F E G

Postorder: A C B E G F D

b)

Inorder: A B C D E

Preorder: C B A D E

Postorder: A B E D C

c)

Inorder: A B C D E F G H I

Preorder: E C B A D H F G I

Postorder: A B D C G F I H E

2. For each of the following key sequences create the binary search tree obtained when the keys are

inserted one-by-one in the order given into an initially empty tree:

(1) 4, 3, 1, 11, 5, 9, 2, 6, 15, 12.

(2) 12, 7, 1, 3, 2, 5, 10, 8, 6, 9.

Give the preorder, inorder, postorder,and level-order traversals of the created binary trees.

In (1) delete keys 2, 3 and 11. In (2) delete keys 5, 6 and 7. After each time of deleting, give the above

traversals.

(1):

- preorder: 4 3 1 2 11 5 9 6 15 12

- inorder: 1 2 3 4 5 6 9 11 12 15

- postorder: 2 1 3 6 9 5 12 15 11 4

- level-order traversals: 4 3 11 1 5 15 2 9 12 6

(1) delete key 2:

- preorder: 4 3 1 11 5 9 6 15 12

- inorder: 1 3 4 5 6 9 11 12 15

- postorder: 1 3 6 9 5 12 15 11 4

- level-order traversals: 4 3 11 1 5 15 9 12 6

(1) delete key 2, 3:

- preorder: 4 1 11 5 9 6 15 12

- inorder: 1 4 5 6 9 11 12 15

- postorder: 1 6 9 5 12 15 11 4

- level-order traversals: 4 1 11 5 15 9 12 6

(1) delete key 2, 3 and 11:

- preorder: 4 1 12 5 9 6 15

- inorder: 1 4 5 6 9 12 15

- postorder: 1 6 9 5 15 12 4

- level-order traversals: 4 1 12 5 15 9 6

(2):

- preorder: 12 7 1 3 2 5 6 10 8 9

- inorder: 1 2 3 5 6 7 8 9 10 12

- postorder: 2 6 5 3 1 9 8 10 7 12

- level-order traversals: 12 7 1 10 3 8 2 5 9 6

(2) delete 5:

- preorder: 12 7 1 3 2 6 10 8 9

- inorder: 1 2 3 6 7 8 9 10 12

- postorder: 2 6 3 1 9 8 10 7 12

- level-order traversals: 12 7 1 10 3 8 2 6 9

(2) delete 5, 6:

- preorder: 12 7 1 3 2 10 8 9

- inorder: 1 2 3 7 8 9 10 12

- postorder: 2 3 1 9 8 10 7 12

- level-order traversals: 12 7 1 10 3 8 2 9

(2) delete 5, 6, 7:

- preorder: 12 1 3 2 10 8 9

- inorder: 1 2 3 8 9 10 12

- postorder: 2 3 1 9 8 10 12

- level-order traversals: 12 1 3 2 10 8 9

3. For each of the key sequences in question 2 create the AVL tree obtained when the keys are inserted

one-by-one in the order given into an initially empty tree.

Give the preorder, inorder, postorder,and level-order traversals of the created binary trees.

(1):

- preorder: 4 2 1 3 9 5 6 12 11 15

- inorder: 1 2 3 4 5 6 9 11 12 15

- postorder: 1 3 2 6 5 11 15 12 9 4

- level-order traversals: 4 2 9 1 3 5 12 6 11 15

(2):

- preorder: 7 5 2 1 3 6 9 8 10 12

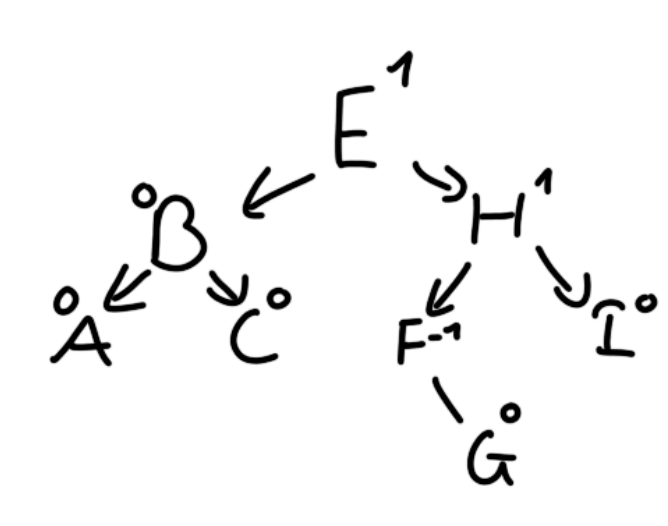
- inorder: 1 2 3 5 6 7 8 9 10 12

- postorder: 1 3 2 6 5 8 12 10 9 7

- level-order traversals: 7 5 9 2 6 8 10 1 3 12

4. The tree (c) in question 1 is height balanced (AVL). Delete the node D and perfom necessary operations so that the tree is still AVL after deletion.

Rotation right at node B and C

=>

5.

a. There are at most 2n + 1 nodes

b. There are at most ⌈n/2h+1⌉ nodes of height h

6. It’s not possible because 57 is less than 63 but it’s was on the right side of 63, 68

