MilenaKuznetsova problem set 7

Solutions by **Milena Kuznetsova**. m.kuznetsova@innopolis.university

---

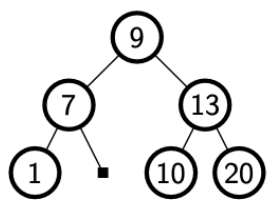
Problem 1

Insert the following keys into an initially empty AVL tree:

$$2, 3, 29, 5, 11, 23, 13, 17, 19, 7$$

For each insertion:

- show the *state of the tree* **after** the insertion
- specify the *number of rotations* performed during the insertion
  Depict each tree using the array representation for binary trees. For example, consider the following AVL tree:



The tree above must be depicted as the following array:

$$|9|7|13|1|-|10|20$$

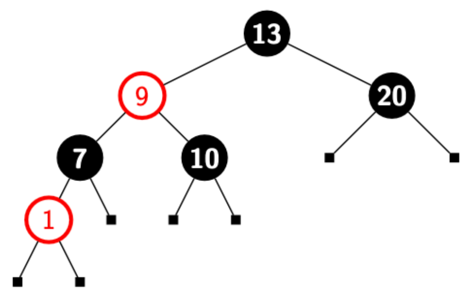
Answer:

Step	State	Rotations
2	`2`	0
3	`2 - 3`	0
29	`3 2 29`	1
5	`3 2 29 - - 5 -`	0
11	`3 2 11 - - 5 29`	2
23	`11 3 29 2 5 23 -`	1

Step	State	Rotations
13	`11 3 23 2 5 13 29`	1
17	`11 3 23 2 5 13 29 - - - - - 17 - -`	0
19	`11 3 23 2 5 17 29 - - - - 13 19 - -`	1
7	`11 3 23 2 5 17 29 - - - 7 13 19 - -`	0

Problem 2

Perform the following operations on the given red-black tree:



- insert 18, 5, 7
- insert 25, 20, 24
- insert 23, 22, 21
- delete 5, 9, 23
- delete 22, 31, 1

For each step:

- show the *state of the tree* **after** the operations
- specify the total *number of rotations* performed during the operations
  Depict each tree using the array representation for binary trees. You must color the keys for **red** nodes with red and keys for **black** nodes with black color. For example, the initial red-black tree presented above is depicted as the following array:

$$13|{\color{red}9}|20|7|10|-|-|{\color{red}1}|-|-|-|-|-|-|-$$

Answer:

Operation	State	Rotations
insert 18, 5, 7	**9 5 13 1 7 10 20 - - - 7 - - 18 -**	3
insert 25, 20, 24	**9 5 13 1 7 10 20 - - - 7 - - 18 24 - - - - 20 25**	2
insert 23, 22, 21	**9 5 20 1 7 13 24 - - - 7 10 18 22 25 - - - - - - 20 23 - - - 21 - -**	3
delete 5, 9, 23	**20 7 24 7 13 21 25 1 - 10 18 20 22 - -**	4
delete 22, 31, 1	**20 7 24 7 13 21 25 - - 10 18 20 - - -**	0

Problem 3

Compare randomly-built and randomly chosen binary search trees for size $n = 4$:

(a) Write down the number of distinct shapes for a binary search trees of size $n = 4$.
Answer: 14

(b) What is the average height of a randomly chosen binary search tree of size n = 4?
Answer: 2.17

(c) Consider $a < b < c < d$. For every permutation of a, b, c, d, write down an array representation of a corresponding binary search tree built from that permutation (by inserting keys in the given order).

- `[a b c d]`
- `[a c d b]`
- `[b a c d]`
- `[b c d a]`
- `[c a b d]`
- `[c b a d]`
- `[c d a b]`
- `[d a b c]`
- `[d c a b]`
- `[d c b a]`

(d) What is the average height of a randomly built binary search tree of size $n = 4$?
Answer: 2.67

(e) Explain, in your own words, why we should not start with a complete binary tree of size n (for any n) with “empty” nodes (i.e. each node does not have any key) and then populate it with

given keys $k_1, k_2, \ldots, k_n$ to achieve the optimal height of the resulting binary search
Answer:

- **BST requires for keys in the left subtree to be smaller than keys in the right subtree. When we start with a complete tree, we might violate this rule because the height of the tree will be predetermined.**
- **The order of insertion matters in BST. Thus, the dynamical insertion of keys in BST will be violated if we create an empty complete tree before inserting.**