Design and Analysis of Algorithms

Lecture-17

Dharmendra Kumar (Associate Professor)
Department of Computer Science and Engineering
United College of Engineering and Research,
Prayagraj

Deletion operation

Suppose we want to delete a node z from a red-black tree T. We use the following steps for this purpose:-

1. First we delete node z using binary search tree deletion process.
2. Find node y in the following way:-
   - If node z has two children then y will be successor of z otherwise y will be z.
3. After finding y, we find x in the following way:-
   - If node y has left child then x will be left child of y otherwise x will be right child of y.
4. If color of y is **red**, then we terminate the process.
5. If color of y is **black**, then we maintain the properties of red-black tree in the following way:-

Deletion operation

(5-a) We start and continue a loop if x is not root node and color of x is **black**.

(5-b) if x is the left child then we do the following actions:-

- (i) Find sibling of x. Let it is denoted by w.
- (ii) There will be four cases:-

  Case-1: If color of w is **red**, then we do the following actions:-

  - (1) w.color = **black**
  - (2) x.p.color = **red**
  - (3) Apply left rotation at parent of node x.

  Case-2: If color of w is **black** and color of its both children is also **black**, then we do the following actions:-

  - (1) w.color = **red**
  - (2) x = x.p

Deletion operation

Case-3: If color of w is **black** and color of its left child is **red** and color of its right child is **black**, then we do the following actions:-

(1) w.left.color = **black**
(2) w.color = **red**
(3) Apply right rotation at node w.

Case-4: If color of w is **black** and color of its right child is **red**, then we do the following actions:-

(1) w.right.color = **black**
(2) w.color = x.p.color
(3) x.p.color = **black**
(4) Apply left rotation at parent of node x.
(5) x = T.root

Deletion operation

(5-c) if x is the right child then we do the following actions:-

(i) Find sibling of x. Let it is denoted by w.

(ii) There will be four cases:-

Case-1: If color of w is **red**, then we do the following actions:-

(1) w.color = **black**

(2) x.p.color = **red**

(3) Apply right rotation at parent of node x.

Case-2: If color of w is **black** and color of its both children is also **black**, then we do the following actions:-

(1) w.color = **red**

(2) x = x.p

Deletion operation

Case-3: If color of w is **black** and color of its right child is **red** and color of its left child is **black**, then we do the following actions:-

(1) w.right.color = **black**

(2) w.color = **red**

(3) Apply left rotation at node w.

Case-4: If color of w is **black** and color of its left child is **red,** then we do the following actions:-

(1) w.left.color = **black**

(2) w.color = x.p.color

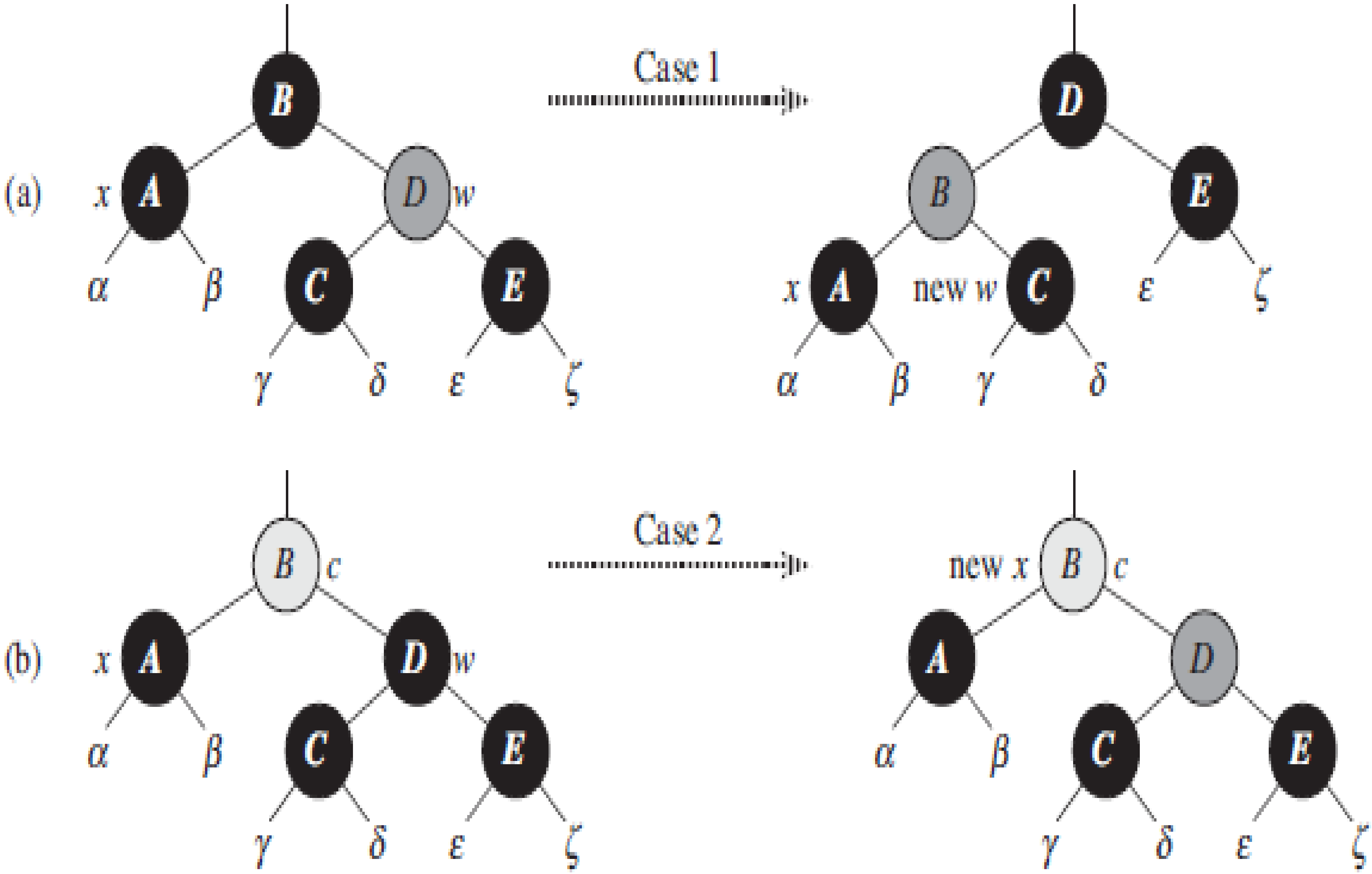
(3) x.p.color = **black**

(4) Apply right rotation at parent of node x.

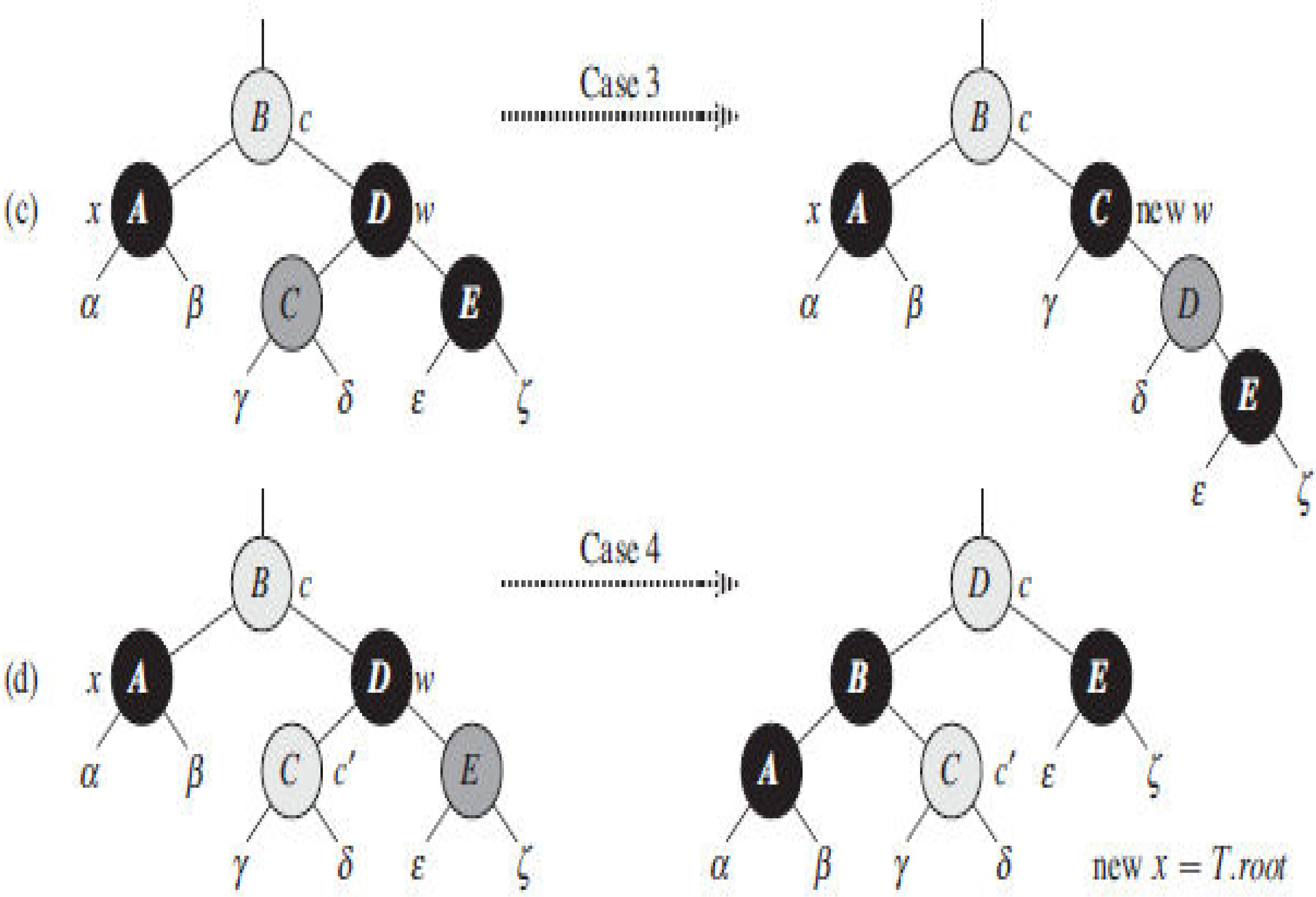
(5) x = T.root

(5-d) After exit from the loop, we make the color of node x to the **black** i.e. x.color = **black**

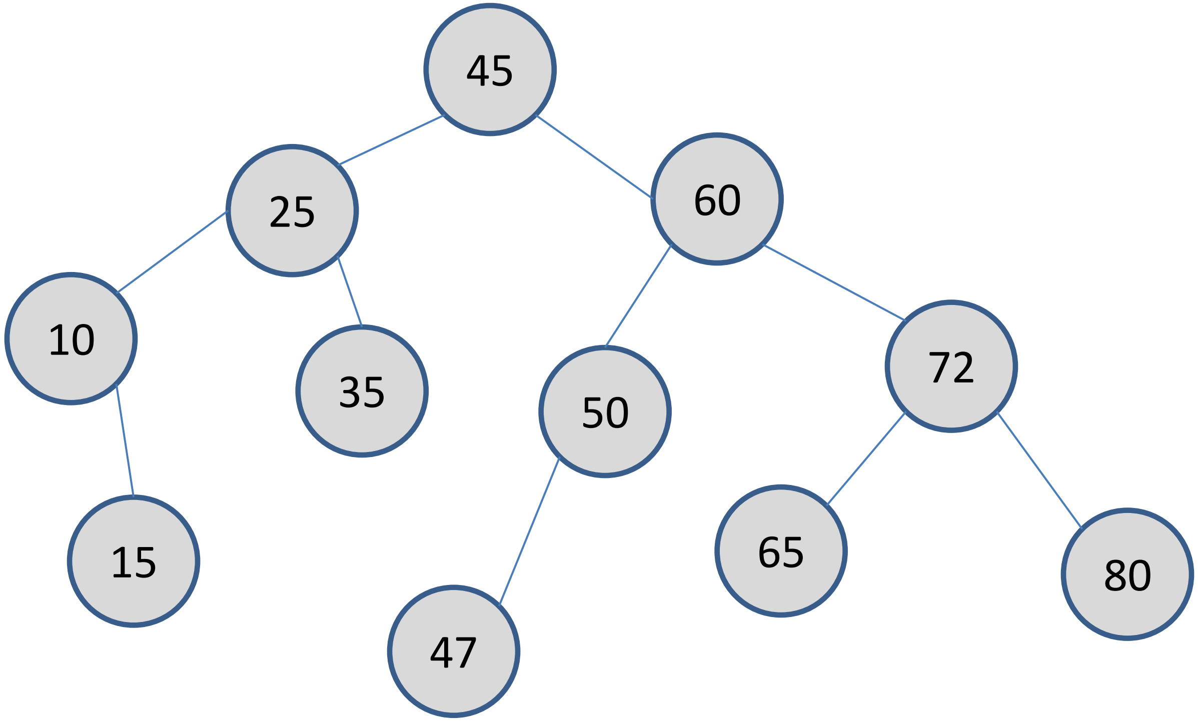
Deletion operation



Deletion operation

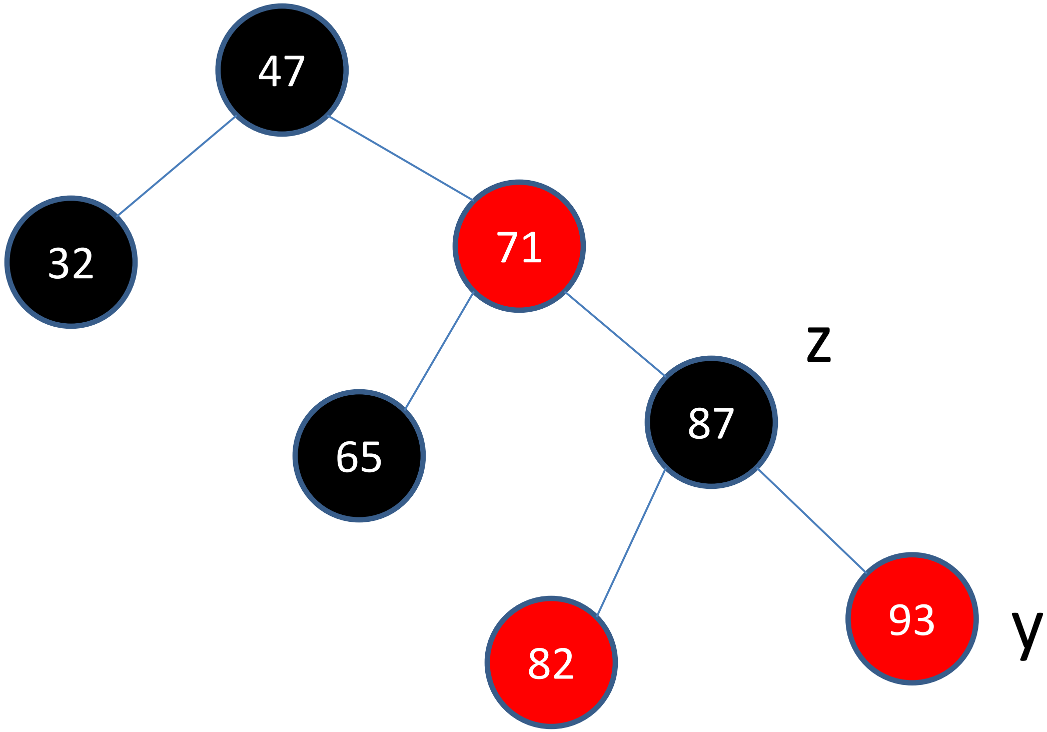


Binary search tree deletion



Deletion operation

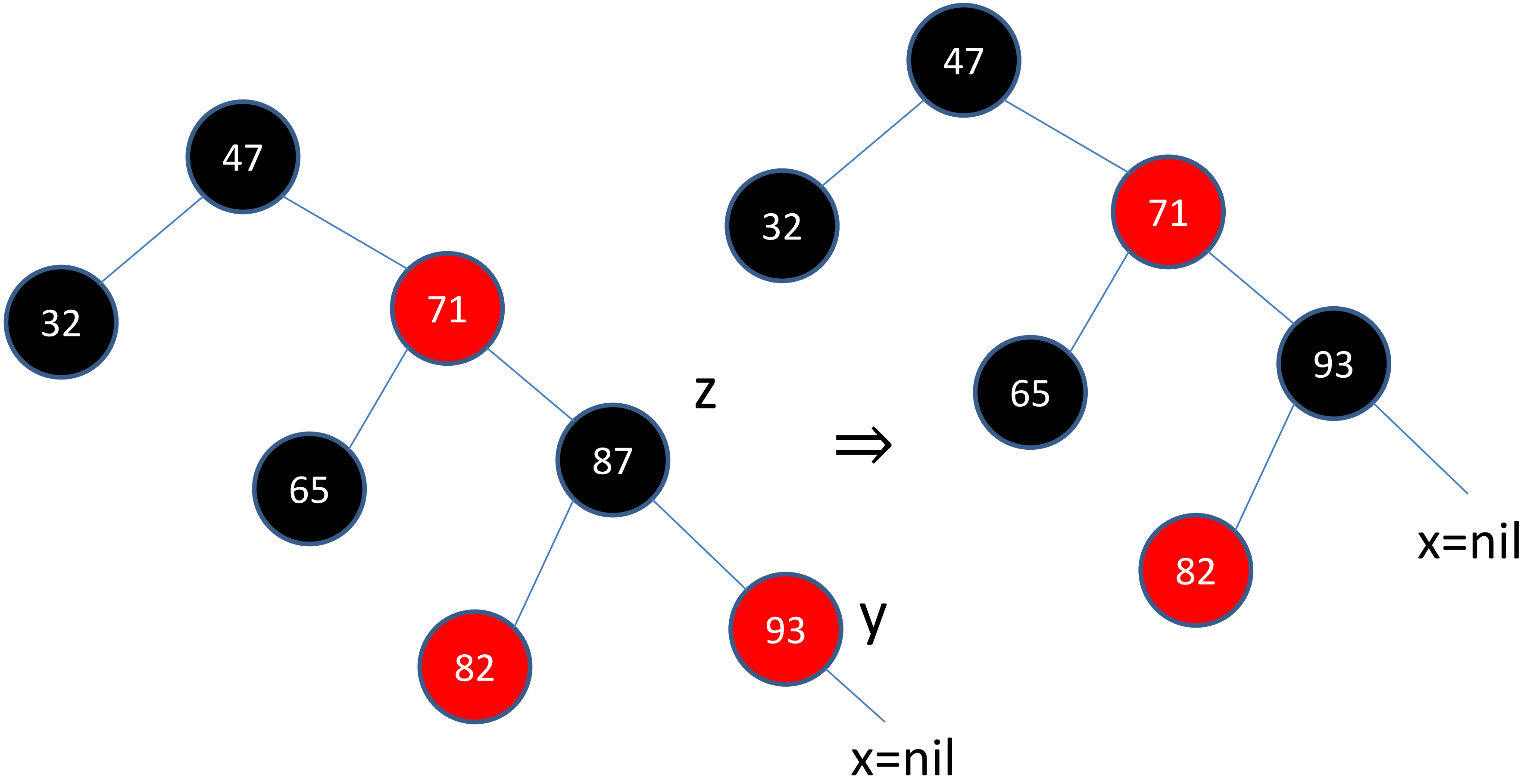
Example: Consider the following red-black tree



Delete the element 87, 32 and 71 in order.

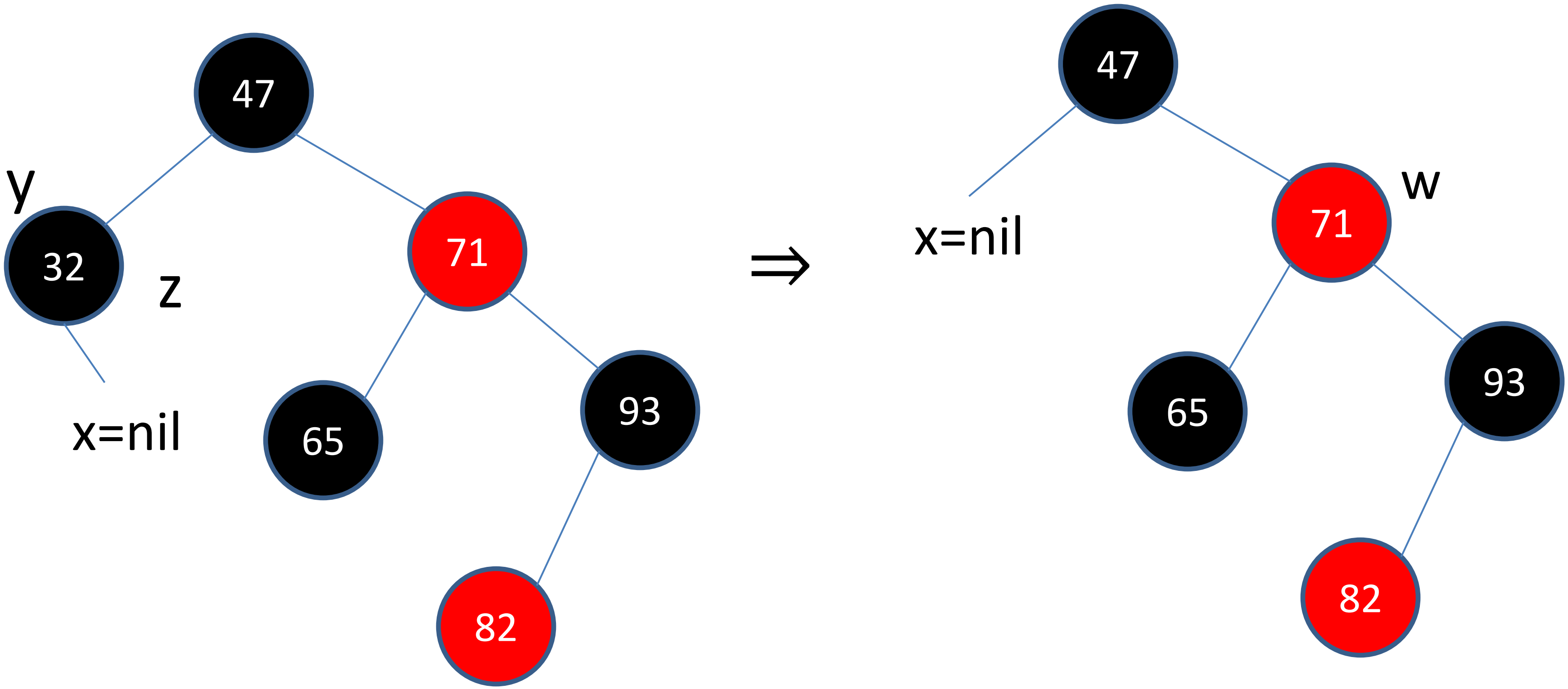
Deletion operation

Deletion of 87



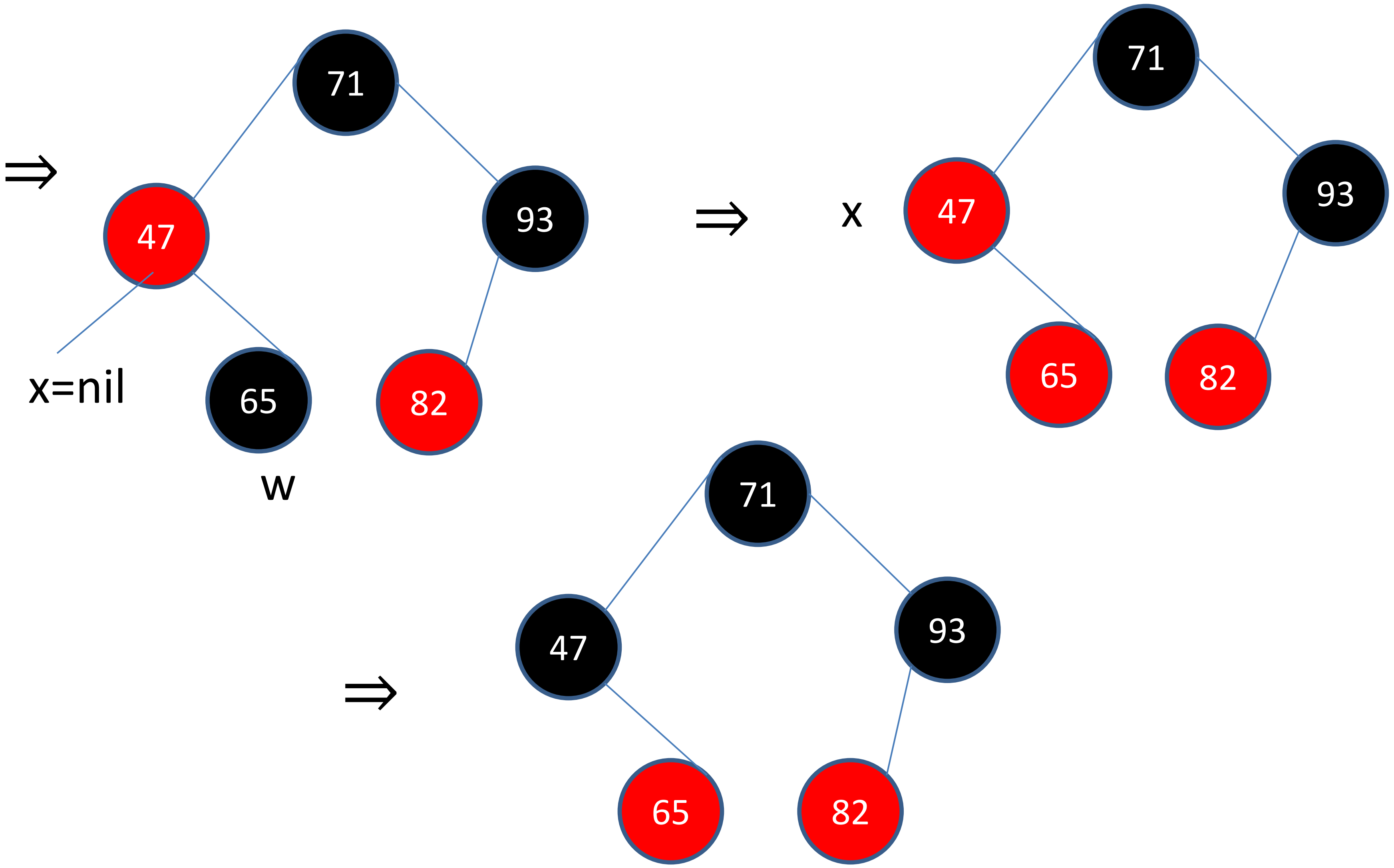
Deletion operation

Deletion of 32



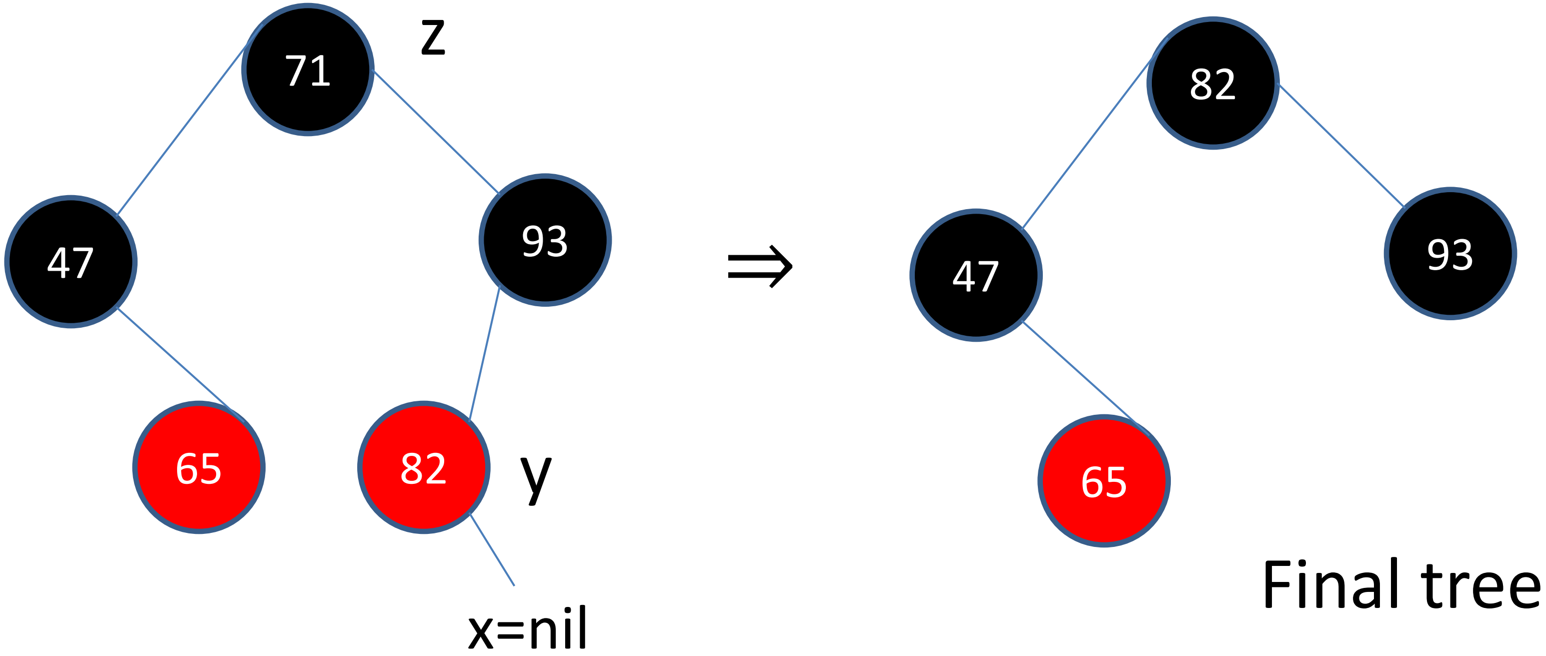
Deletion operation

Deletion of 32(continue)



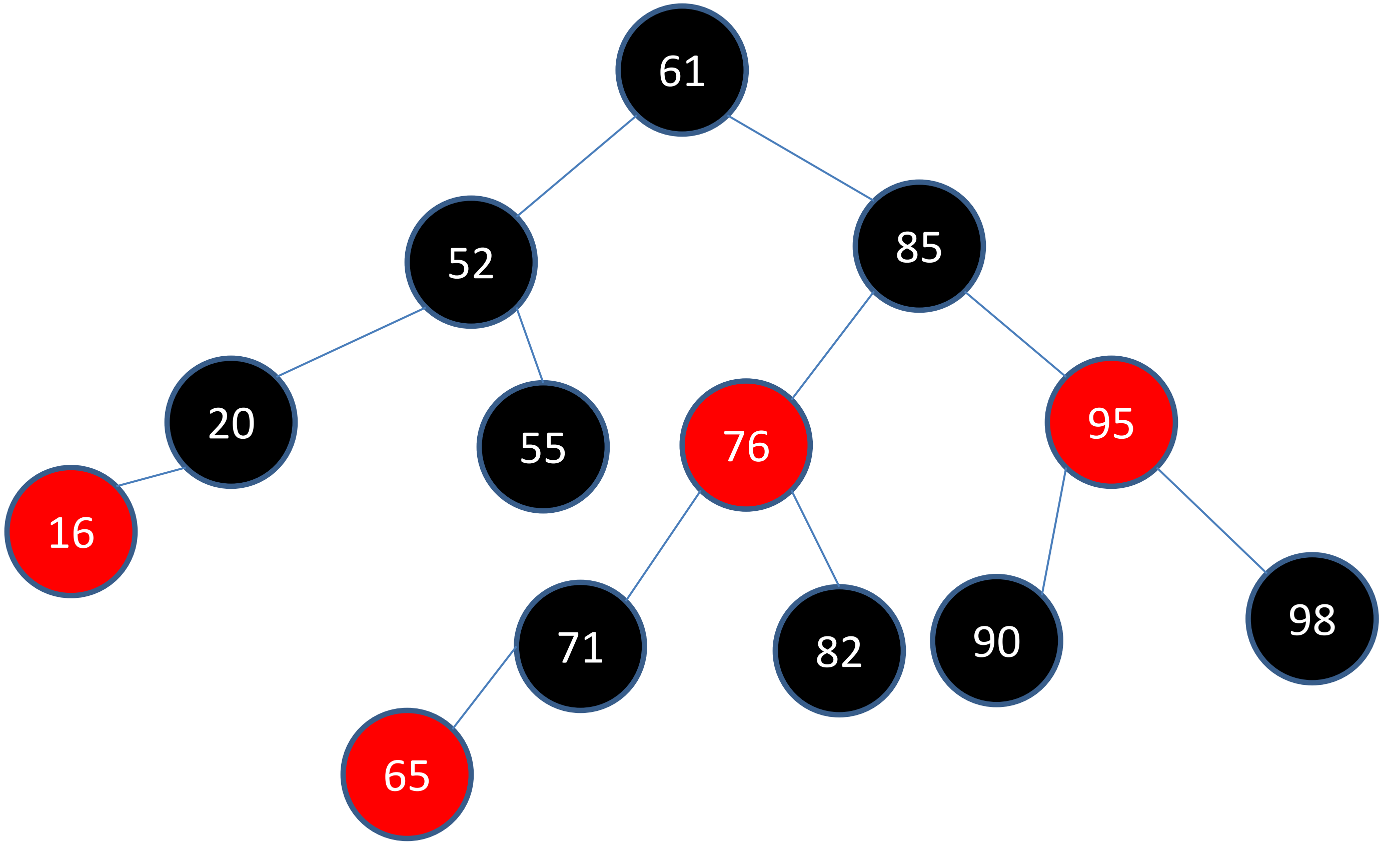
Deletion operation

Deletion of 71



Deletion operation

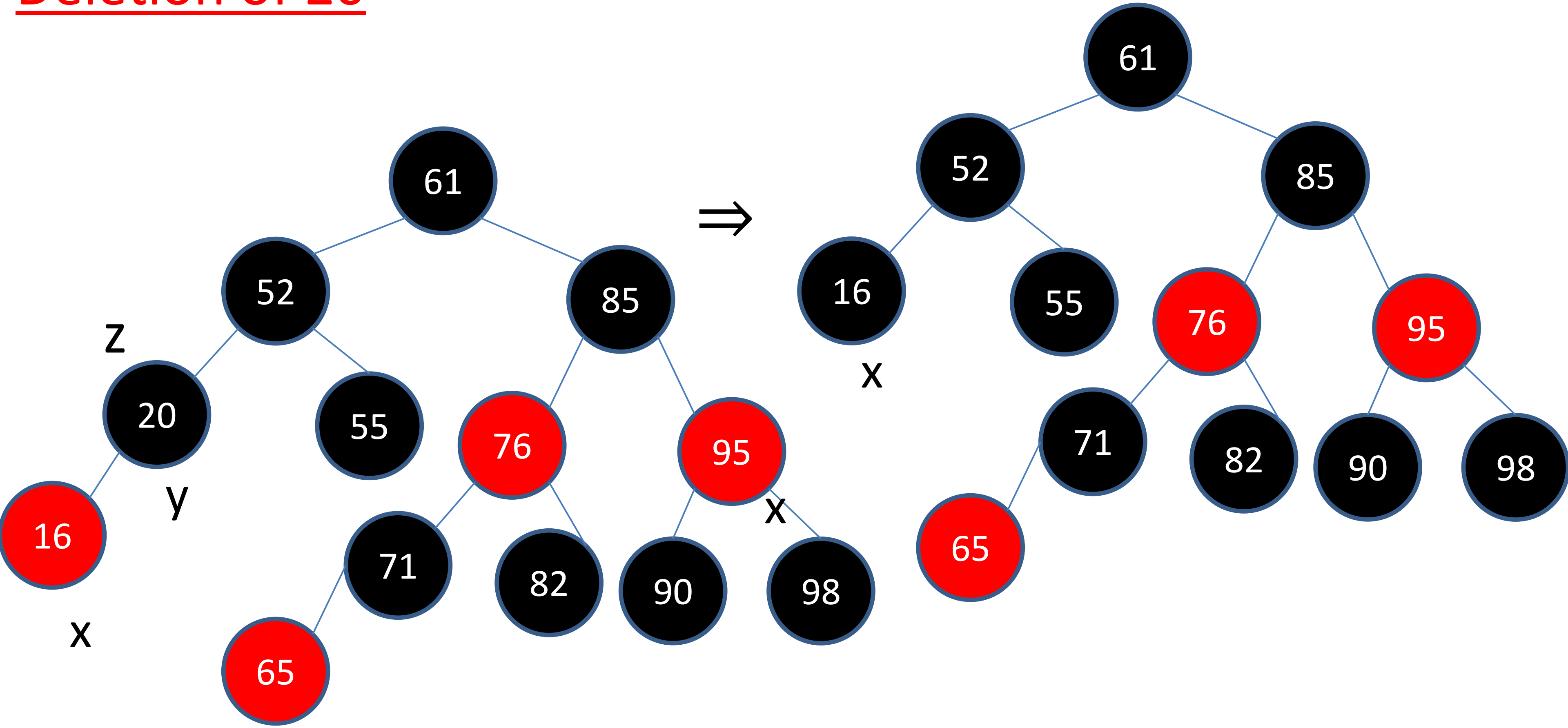
Example: Consider the following red-black tree



Delete the element 20, 85, 95 and 98 in order.

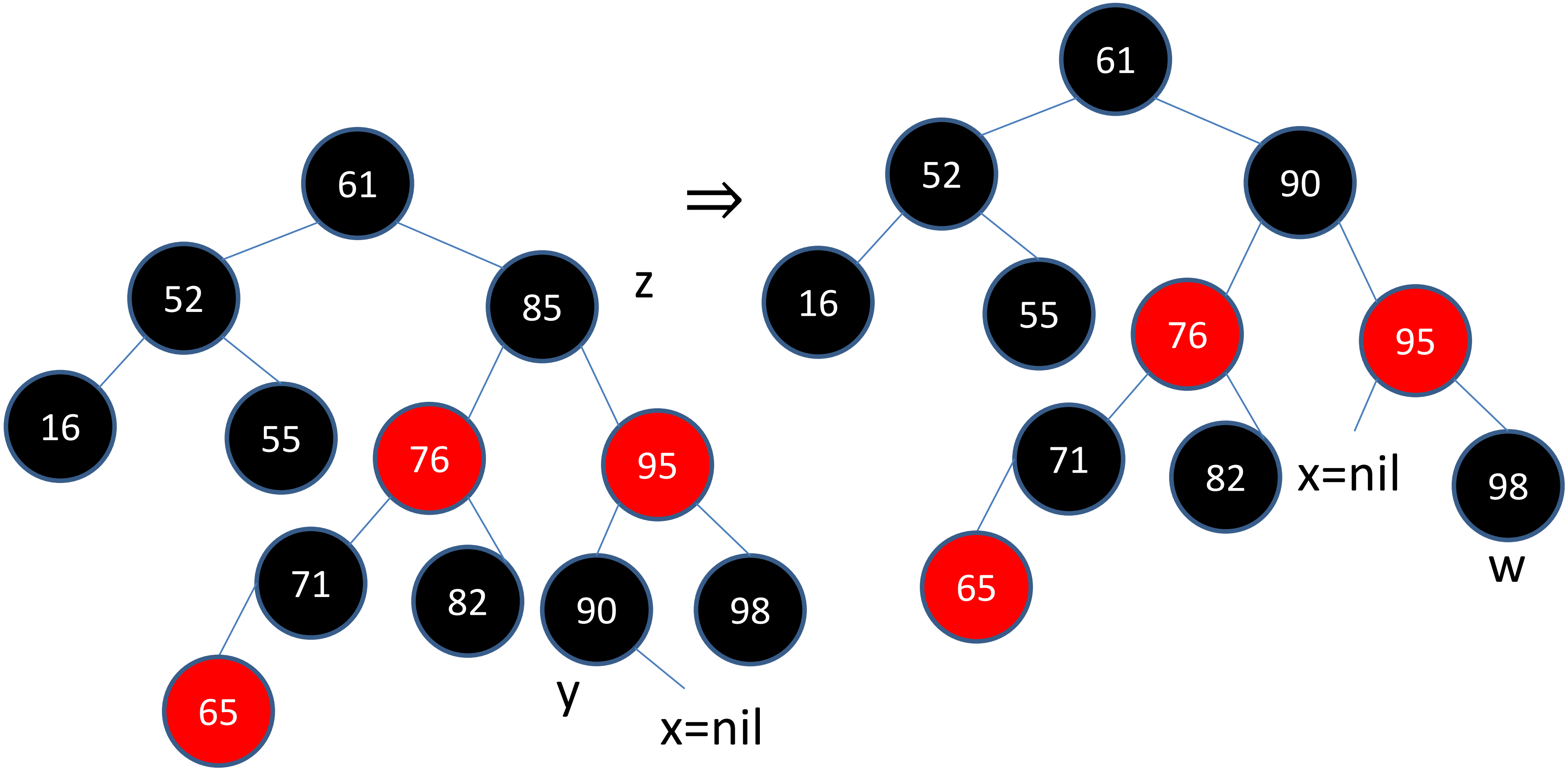
Deletion operation

Deletion of 20



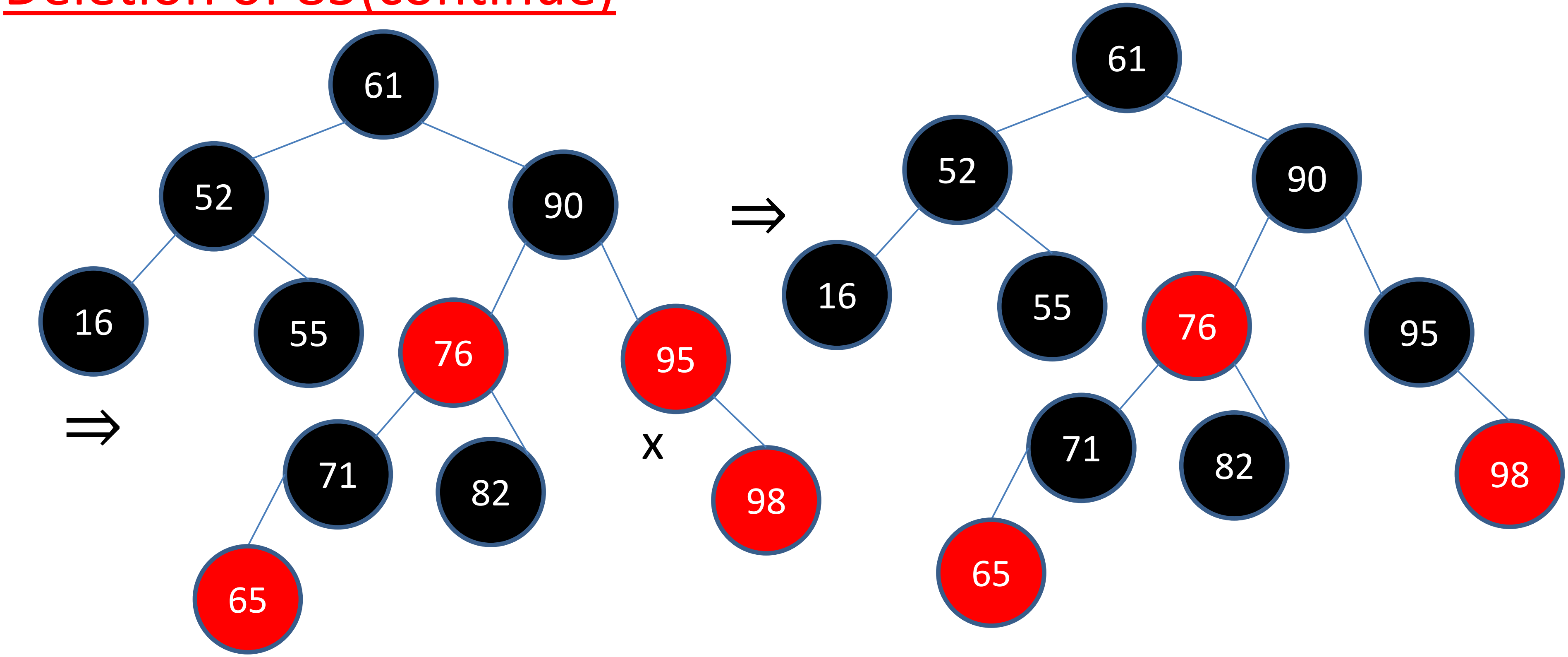
Deletion operation

Deletion of 85



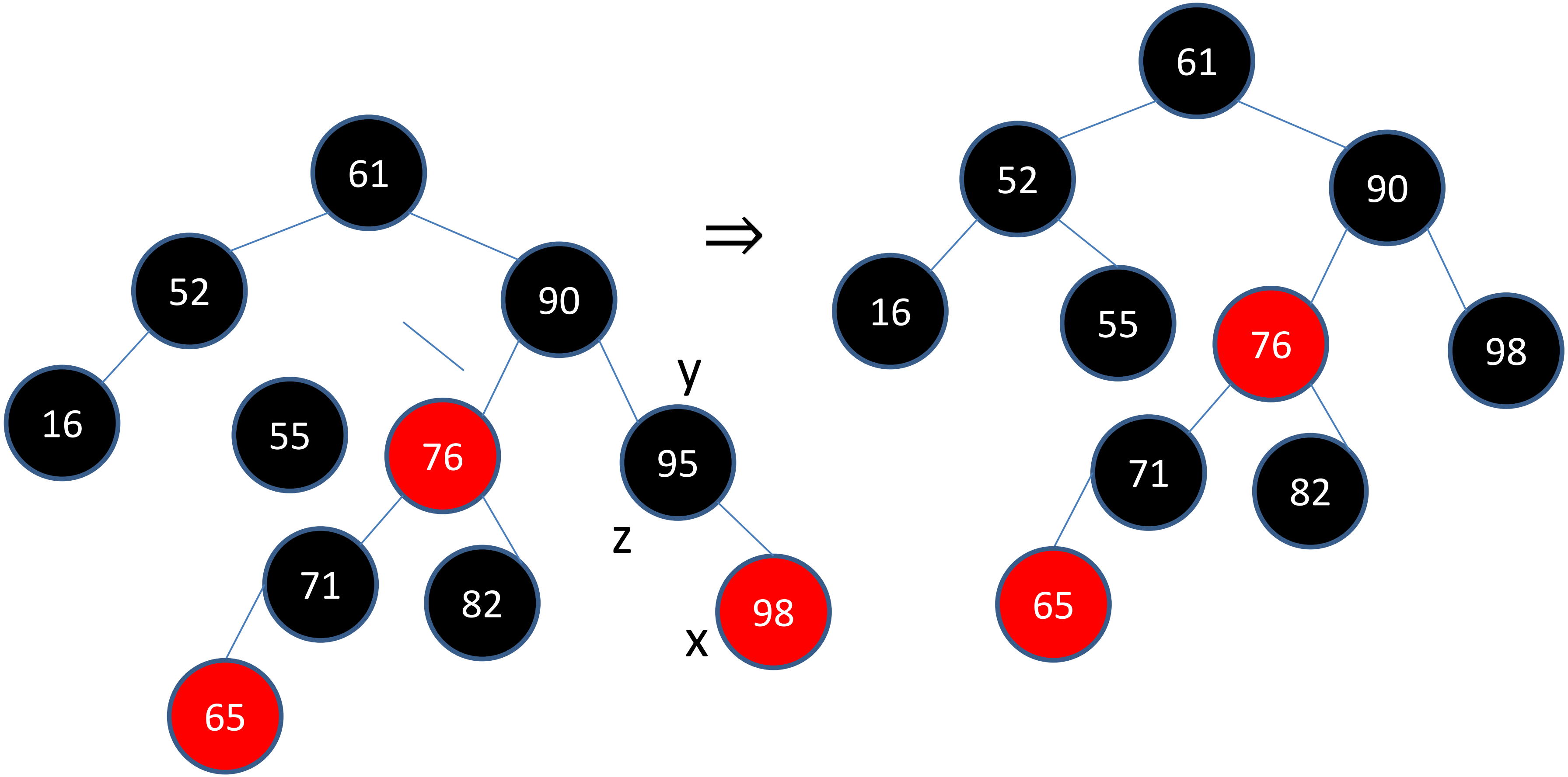
Deletion operation

Deletion of 85(continue)



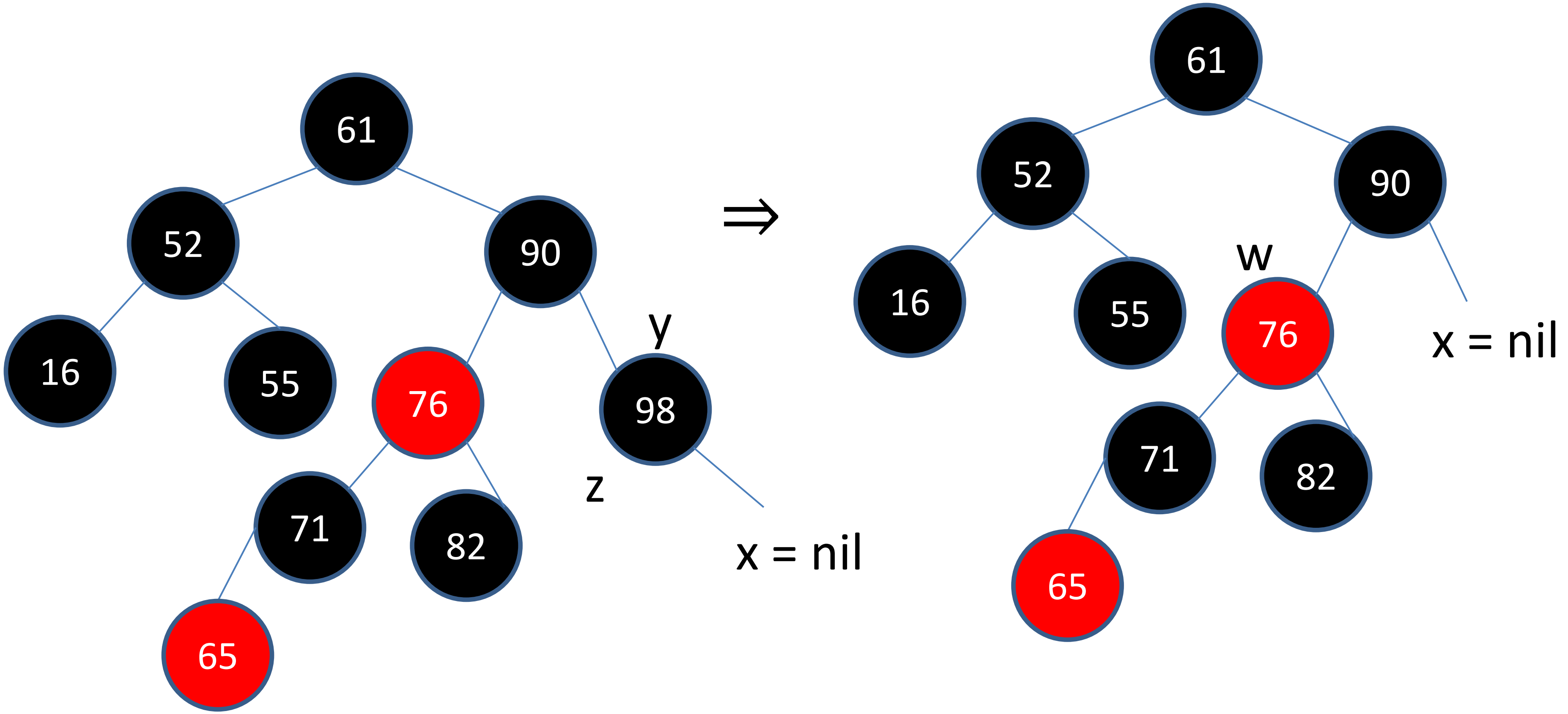
Deletion operation

Deletion of 95



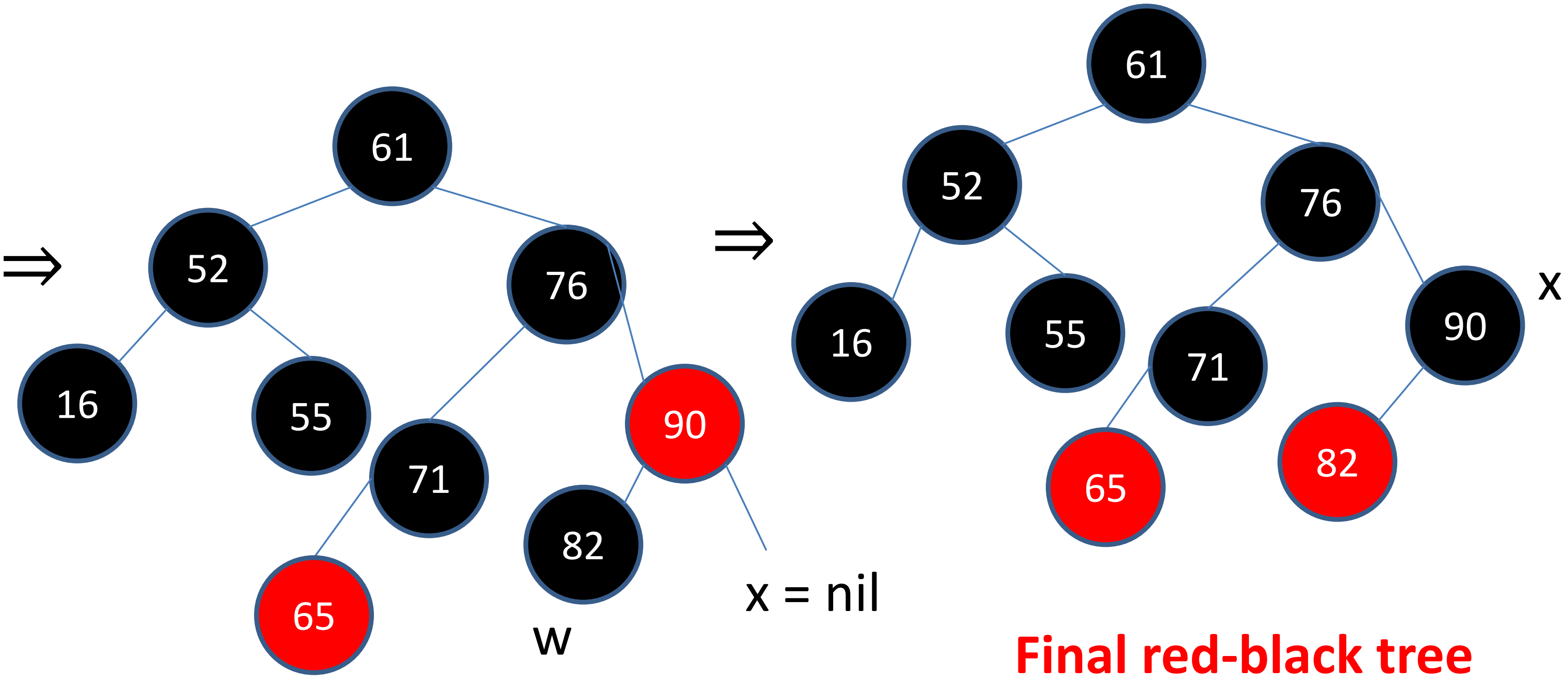
Deletion operation

Deletion of 98



Deletion operation

Deletion of 98(continue)



Final red-black tree

AKTU Examination Questions

1. Insert the elements 8, 20, 11, 14, 9, 4, 12 in a Red-Black Tree and delete 12, 4, 9, 14 respectively.
2. What is Red-Black tree? Write an algorithm to insert a node in an empty red-black tree explain with suitable example.
3. Insert the following element in an initially empty RB-Tree. 12, 9, 81, 76, 23, 43, 65, 88, 76, 32, 54. Now Delete 23 and 81.
4. Write the properties of Red-Black Tree. Illustrate with an example, how the keys are inserted in an empty red-black tree.

Thank You.