

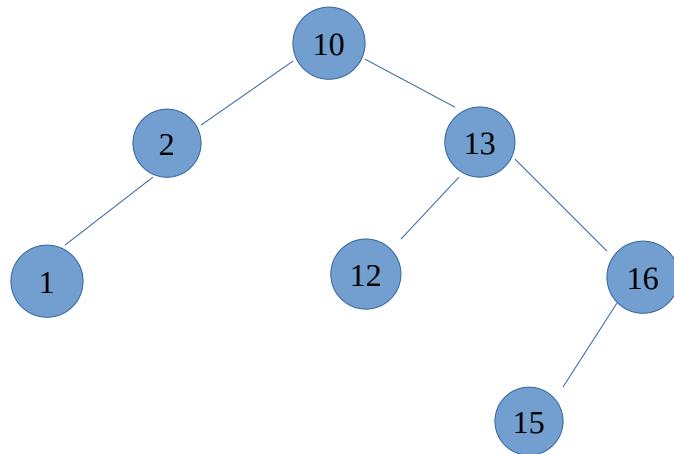
```

node* delete_data(int data, node* root){
    if(root == NULL) return NULL;
    if(data > root->data)
        root->right = delete_data(data, root->right);
    else if(data < root->data)
        root->left = delete_data(data, root->left);
    else{
        if(root->left == NULL) return root->right;
        if(root->right == NULL) return root->left;

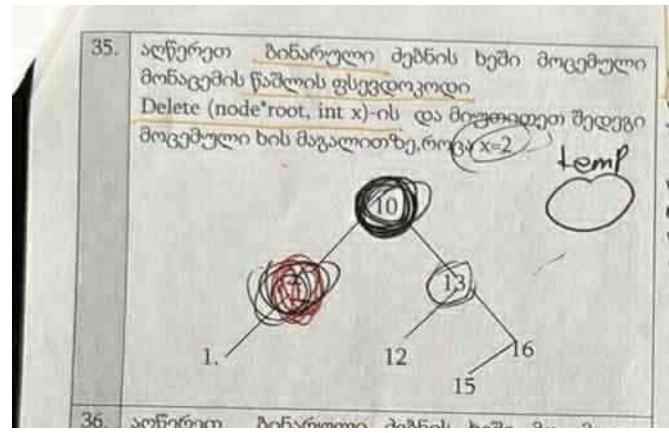
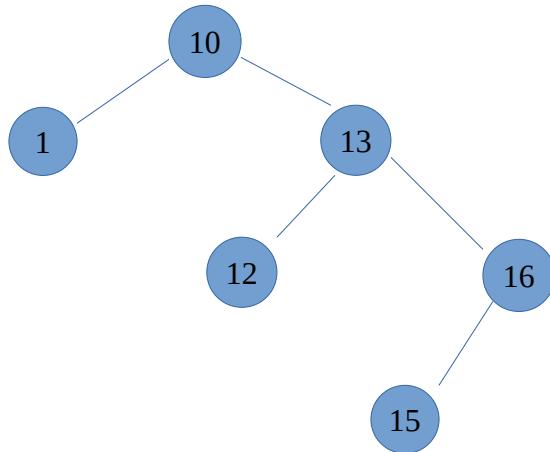
        root->data = search_min(root->right)->data;
        root->right = delete_min(root->right);
    }
    return root;
}

```

მოცემული ხე:

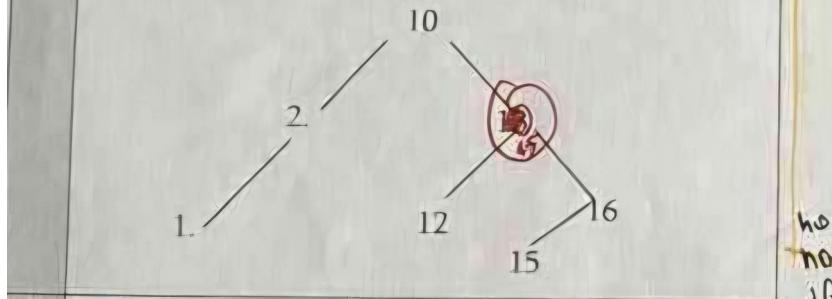


$x=2$ - წაშლის შემდეგ:

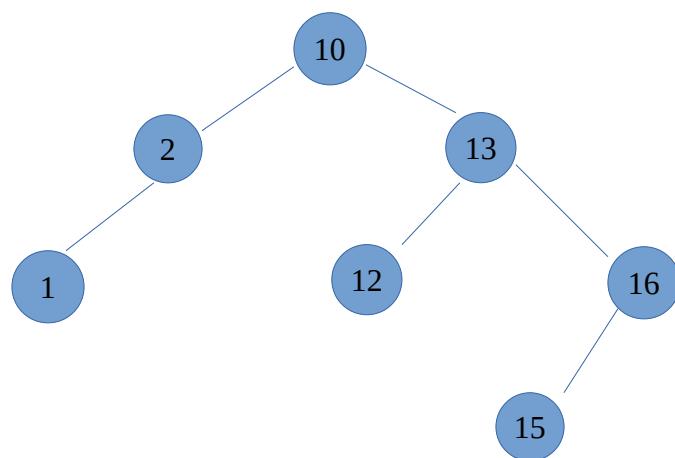


36.

36. აღწერეთ ჩინარული მემნის ხეში მოცემული
მონაცემის წაშლის ფსევდოკოდი
Delete (node *root, int x)-ის და მიუთითეთ შედეგი
მოცემული ხის მაგალითზე, როცა $x=13$



მოცემული ხე:



13 უნდა შეიცვალოს მის მარჯვნივ მყოფი ციფრებიდან ყველაზე მინიმალურით
და ასეთი აღმოჩნდა 15.

$x=13$ - წაშლის შემდეგ:

