

Write-up

Insertion in AVL tree

```
Node * insert(Node* h, int key)
{
    if (h == NULL)
    {
        h = new Node;
        h->data = key;
        h->left = NULL;
        h->right = NULL;
        return h;
    }
    else if (key < h->data)
    {
        h->left = insert(h->left, key);
        h = balance(h);
    }
    else if (key < h->data)
    else if (key >= h->data)
    {
        h->right = insert(h->right, key);
        h = balance(h);
    }
    return h;
}
```

Deletion in AVL tree

```
Node* del(Node* h, int key)
{
```

```
    if (h == NULL)
```

```
        return h;
```

```
    if (key < h->key)
```

```
        h->left = del(h->left, key);
```

```
    else if (key >= h->key)
```

```
        h->right = del(h->right, key);
```

```
    else {
```

```
        if ((root->left == NULL) || (root->right == NULL))
        {
```

```
            Node* t = h->left ? h->left :
                               h->right;
```

```
            if (t == NULL)
```

```
            {
```

```
                t = h;
```

```
                h = NULL;
```

```
            }
```

```
            else
```

```
                *h = *t;
```

```
                free(t);
```

```
            }
```

```
        else {
```

```
            Node* t = minNode(h->right);
```

```
            h->key = t->key;
```

```
            h->right = del(h->right, t->key);
```

```
        }
```

```
    }
```

```
}
```