1.Write a c program to find the AVL tree

main.c

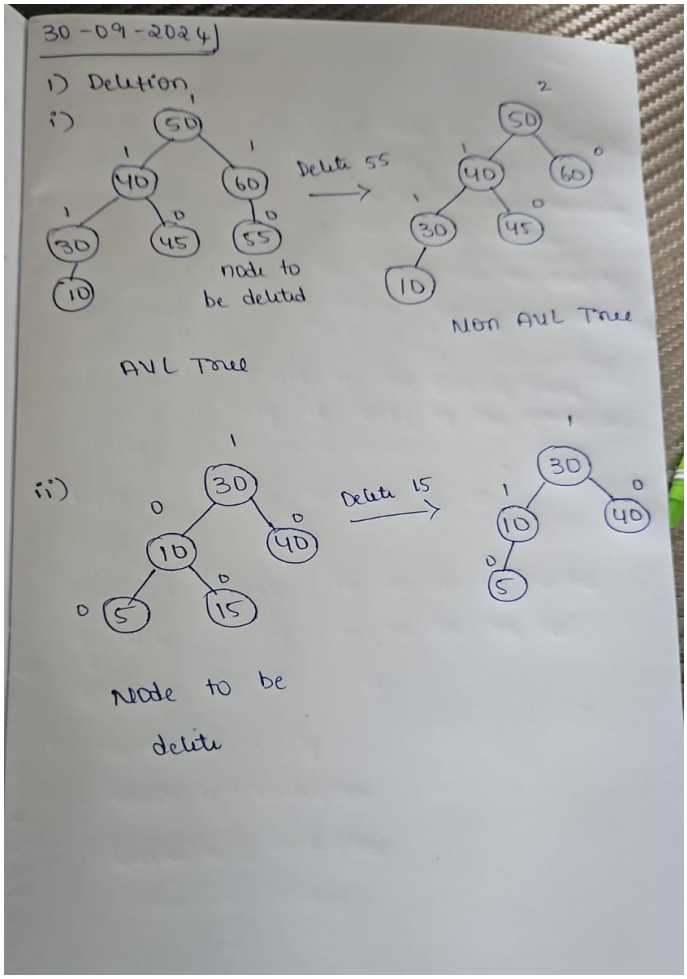
```c
#include <stdio.h>
#include <stdlib.h>
struct Node {
    int key;
    struct Node* left;
    struct Node* right;
    int height;
};
int height(struct Node* N) {
    if (N == NULL)
        return 0;
    return N->height;
}
int max(int a, int b) {
    return (a > b) ? a : b;
}
struct Node* createNode(int key) {
    struct Node* node = (struct Node*)malloc(sizeof(struct Node));
    node->key = key;
    node->left = NULL;
    node->right = NULL;
    node->height = 1;
    return node;
}
struct Node* rightRotate(struct Node* y) {
    struct Node* x = y->left;
    struct Node* T2 = x->right;
    x->right = y;
    y->left = T2;
    y->height = max(height(y->left), height(y->right)) + 1;
    x->height = max(height(x->left), height(x->right)) + 1;
    return x;
}
struct Node* leftRotate(struct Node* x) {
    struct Node* y = x->right;
    struct Node* T2 = y->left;
```

Output

```
/tmp/3CaY3GzJDr.o
In-order traversal of the constructed AVL tree is: 10 20 25 30 40 50

=== Code Execution Successful ===
```

2.Write a problem on deletion operation (AVL Tree)



3.Write a problem on searching operation (AVL Tree)