```
/*Create binary tree and perform recursive traversals*/
#include <iostream>
using namespace std;

// Structure for a node of a binary tree
struct Node {
    int data;
    Node* left;
    Node* right;
};

// Function to create a new node and return its address
Node* getNewNode(int data) {
    Node* newNode = new Node();
    newNode->data = data;
    newNode->left = newNode->right = NULL;
    return newNode;
}

// Recursive function to do pre-order traversal of the binary tree
void preOrder(Node* root) {
    if (root == NULL) return;
    cout << root->data << " ";
    preOrder(root->left);
    preOrder(root->right);
}

// Recursive function to do in-order traversal of the binary tree
void inOrder(Node* root) {
    if (root == NULL) return;
    inOrder(root->left);
    cout << root->data << " ";
    inOrder(root->right);
}

// Recursive function to do post-order traversal of the binary tree
void postOrder(Node* root) {
    if (root == NULL) return;
    postOrder(root->left);
    postOrder(root->right);
    cout << root->data << " ";
}

int main() {
    Node* root = getNewNode(1);
    root->left = getNewNode(2);
    root->right = getNewNode(3);
    root->left->left = getNewNode(4);
    root->left->right = getNewNode(5);

    cout << "Pre-order traversal: ";
    preOrder(root);
    cout << endl;

    cout << "In-order traversal: ";
    inOrder(root);
    cout << endl;

    cout << "Post-order traversal: ";
    postOrder(root);
    cout << endl;

    return 0;
}
```