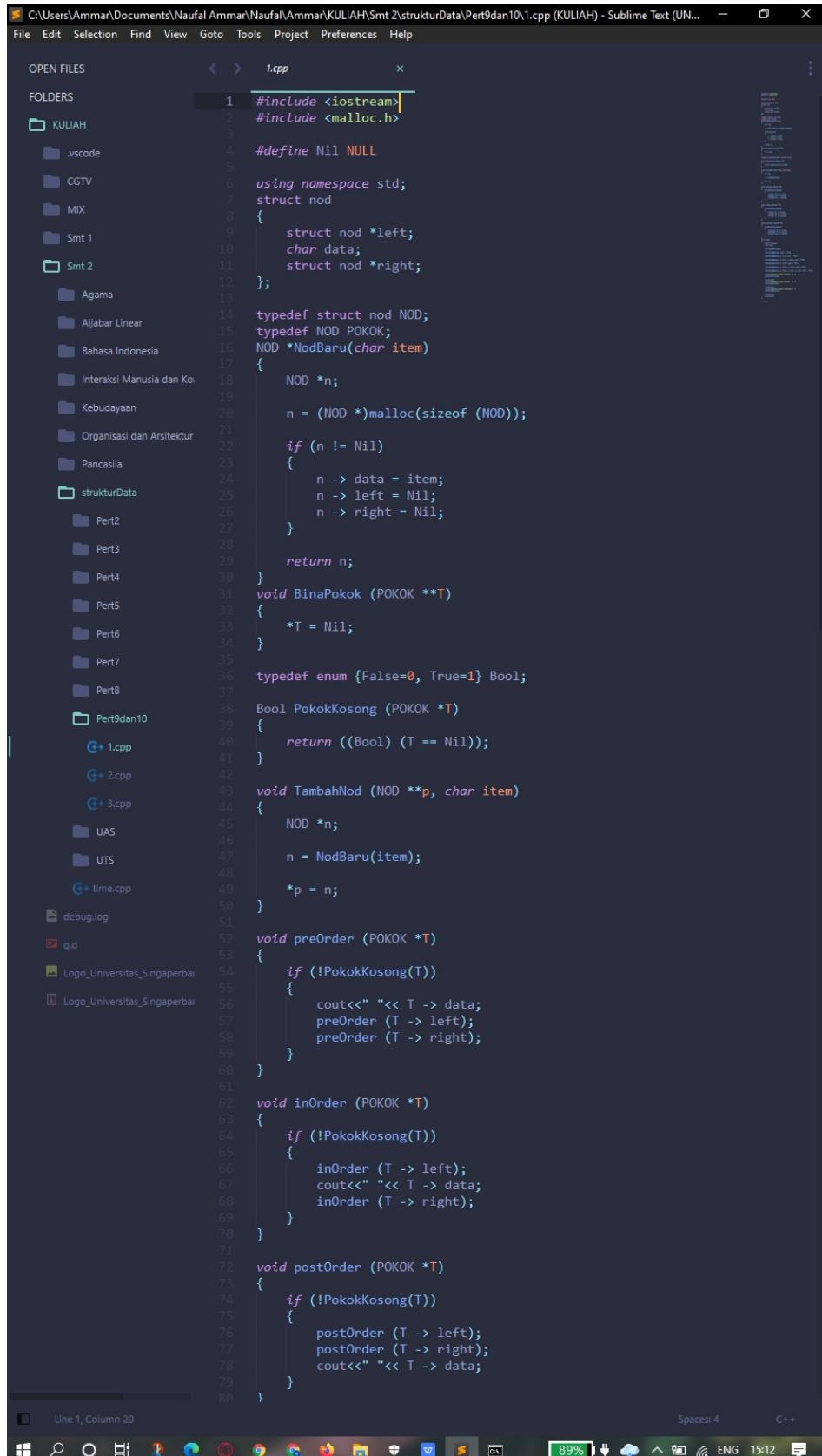


Naufal Ammar Hidayatulloh

2E - Teknik Informatika

2010631170104



The screenshot shows a Sublime Text window displaying a C++ program named `1.cpp`. The code defines a binary tree structure with nodes containing character data and pointers to left and right children. It includes functions for creating new nodes, checking if a tree is empty, inserting nodes, and performing pre-order, in-order, and post-order traversals. The code uses standard library headers like `<iostream>` and `<malloc.h>`, and defines a `Nil` constant.

```
#include <iostream>
#include <malloc.h>

#define Nil NULL

using namespace std;
struct nod
{
    struct nod *left;
    char data;
    struct nod *right;
};

typedef struct nod NOD;
typedef NOD POKOK;
NOD *NodBaru(char item)
{
    NOD *n;
    n = (NOD *)malloc(sizeof(NOD));
    if (n != Nil)
    {
        n->data = item;
        n->left = Nil;
        n->right = Nil;
    }
    return n;
}
void BinaPokok (POKOK **T)
{
    *T = Nil;
}

typedef enum {False=0, True=1} Bool;
Bool PokokKosong (POKOK *T)
{
    return ((Bool) (T == Nil));
}

void TambahNod (NOD **p, char item)
{
    NOD *n;
    n = NodBaru(item);
    *p = n;
}

void preOrder (POKOK *T)
{
    if (!PokokKosong(T))
    {
        cout<<" "<< T->data;
        preOrder (T->left);
        preOrder (T->right);
    }
}

void inOrder (POKOK *T)
{
    if (!PokokKosong(T))
    {
        inOrder (T->left);
        cout<<" "<< T->data;
        inOrder (T->right);
    }
}

void postOrder (POKOK *T)
{
    if (!PokokKosong(T))
    {
        postOrder (T->left);
        postOrder (T->right);
        cout<<" "<< T->data;
    }
}
```

C:\Users\Ammar\Documents\Naufal Ammar\Naufal Ammar\KULIAH\Smt 2\strukturData\Pert9dan10\1.cpp (KULIAH) - Sublime Text (UN...)

File Edit Selection Find View Goto Tools Project Preferences Help

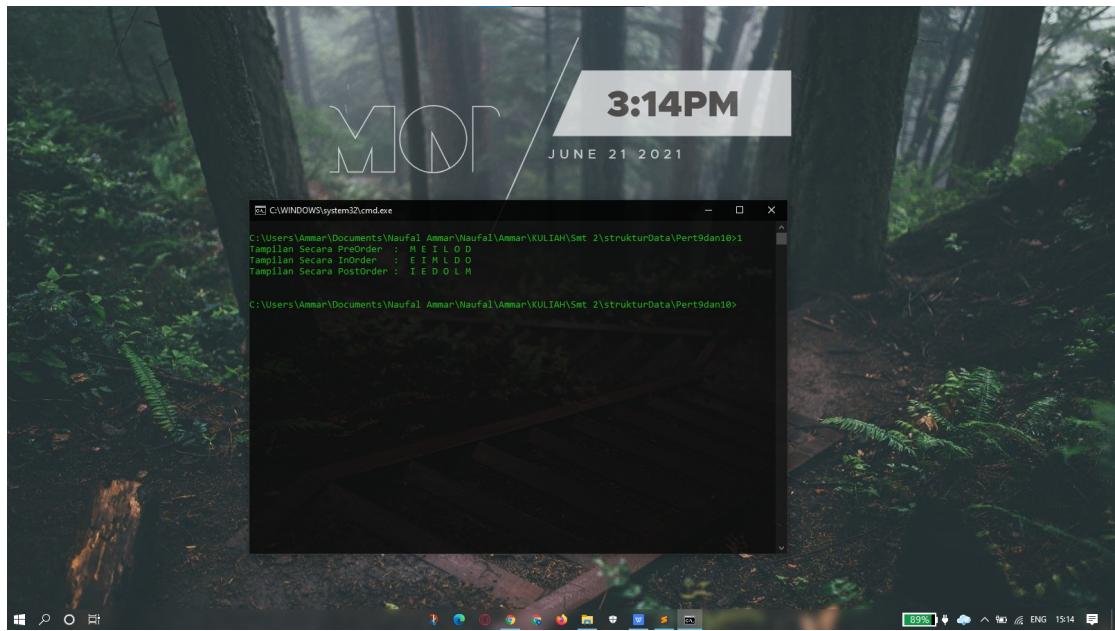
OPEN FILES

FOLDERS

```
41 }  
42  
43 void TambahNod (NOD **p, char item)  
44 {  
45     NOD *n;  
46  
47     n = NodBaru(item);  
48  
49     *p = n;  
50 }  
51  
52 void preOrder (POKOK *T)  
53 {  
54     if (!PokokKosong(T))  
55     {  
56         cout<< " << T -> data;  
57         preOrder (T -> left);  
58         preOrder (T -> right);  
59     }  
60 }  
61  
62 void inOrder (POKOK *T)  
63 {  
64     if (!PokokKosong(T))  
65     {  
66         inOrder (T -> left);  
67         cout<< " << T -> data;  
68         inOrder (T -> right);  
69     }  
70 }  
71  
72 void postOrder (POKOK *T)  
73 {  
74     if (!PokokKosong(T))  
75     {  
76         postOrder (T -> left);  
77         postOrder (T -> right);  
78         cout<< " << T -> data;  
79     }  
80 }  
81 int main()  
82 {  
83     POKOK *kelapa;  
84     char buah;  
85  
86     BinaPokok(&kelapa);  
87  
88     TambahNod(&kelapa, buah = 'M');  
89  
90     TambahNod(&kelapa -> left, buah = 'E');  
91  
92     TambahNod(&kelapa -> left -> right, buah = 'I');  
93  
94     TambahNod(&kelapa -> right, buah = 'L');  
95  
96     TambahNod(&kelapa -> right -> right, buah = 'O');  
97  
98     TambahNod(&kelapa -> right -> right -> left, buah = 'D');  
99  
100    cout<<"Tampilan Secara PreOrder : ";  
101    preOrder(kelapa);  
102  
103    cout<<endl;  
104    cout<<"Tampilan Secara InOrder : ";  
105    inOrder(kelapa);  
106  
107    cout<<endl;  
108    cout<<"Tampilan Secara PostOrder : ";  
109    postOrder(kelapa);  
110  
111    cout<<endl;  
112    cout<<endl;  
113 }  
114  
115 // Modul
```

Line 1, Column 20

Spaces: 4 C++



The screenshot shows a Sublime Text window with the following details:

- Title Bar:** C:\Users\Ammar\Documents\Naufal Ammar\Naufal Ammar\KULIAH\Smt 2\strukturData\Pert9dan10\2.cpp (KULIAH) - Sublime Text (UN...)
- Menu Bar:** File Edit Selection Find View Goto Tools Project Preferences Help
- Left Panel (File Explorer):** Shows a file structure:
 - Smt 1
 - Smt 2
 - Agama
 - Aljabar Linear
 - Bahasa Indonesia
 - Interaksi Manusia dan I
 - Kebudayaan
 - Organisasi dan Arsitekt
 - Pancasila
 - strukturData
 - Pert2
 - Pert3
 - Pert4
 - Pert5
 - Pert6
 - Pert7
 - Pert8
 - Pert9dan10
 - G+ 1.cpp
 - G+ 2.cpp
 - G+ 3.cpp
 - UAS
 - UTS
 - G+ time.cpp
 - debug.log
 - g.d
 - Logo_Universitas_Singaperl
 - Logo_Universitas_Singaperl
- Center Panel (Code Editor):** The file 2.cpp contains the following C++ code:

```
// Tree traversal in C++
#include <iostream>
using namespace std;

struct Node
{
    int data;
    struct Node *left, *right;
    Node(int data)
    {
        this->data = data;
        left = right = NULL;
    }
};

// Preorder traversal
void preorderTraversal(struct Node *node)
{
    if (node == NULL)
        return;

    cout << node->data << "->";
    preorderTraversal(node->left);
    preorderTraversal(node->right);
}

// Postorder traversal
void postorderTraversal(struct Node *node)
{
    if (node == NULL)
        return;

    postorderTraversal(node->left);
    postorderTraversal(node->right);
    cout << node->data << "->";
}

// Inorder traversal
void inorderTraversal(struct Node *node)
{
    if (node == NULL)
        return;

    inorderTraversal(node->left);
    cout << node->data << "->";
    inorderTraversal(node->right);
}

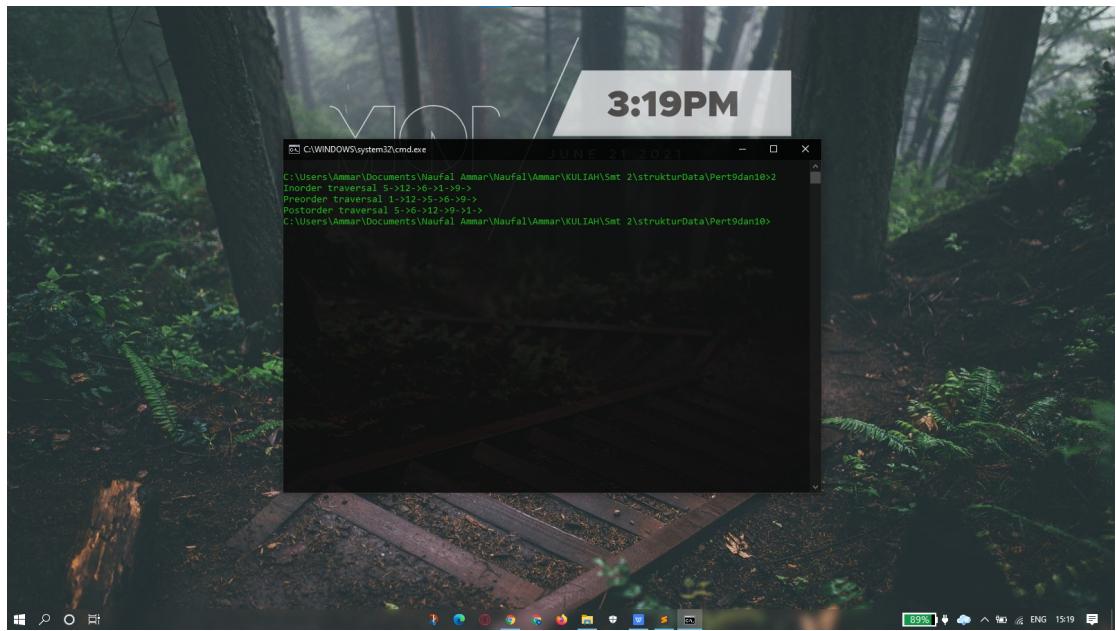
int main()
{
    struct Node *root = new Node(1);
    root->left = new Node(2);
    root->right = new Node(3);
    root->left->left = new Node(4);
    root->left->right = new Node(5);

    cout << "Inorder traversal ";
    inorderTraversal(root);

    cout << "\nPreorder traversal ";
    preorderTraversal(root);

    cout << "\nPostorder traversal ";
    postorderTraversal(root);
}

// Programiz
```
- Bottom Status Bar:** Line 1, Column 25, 99%, Spaces: 4, C++, ENG, 15:18



```
// C++ program for different tree traversals
#include <iostream>
using namespace std;

/* A binary tree node has data, pointer to left child
and a pointer to right child */
struct Node
{
    int data;
    struct Node *left, *right;
    Node(int data)
    {
        this->data = data;
        left = right = NULL;
    }
};

/* Given a binary tree, print its nodes according to the
"bottom-up" postorder traversal. */
void printPostorder(struct Node *node)
{
    if (node == NULL)
        return;

    // first recur on left subtree
    printPostorder(node->left);

    // then recur on right subtree
    printPostorder(node->right);

    // now deal with the node
    cout << node->data << " ";
}

/* Given a binary tree, print its nodes in inorder*/
void printInorder(struct Node *node)
{
    if (node == NULL)
        return;

    /* first recur on left child */
    printInorder(node->left);

    /* then print the data of node */
    cout << node->data << " ";

    /* now recur on right child */
    printInorder(node->right);
}

/* Given a binary tree, print its nodes in preorder*/
void printPreorder(struct Node *node)
{
    if (node == NULL)
        return;

    /* first print data of node */
    cout << node->data << " ";

    /* then recur on left subtree */
    printPreorder(node->left);

    /* now recur on right subtree */
    printPreorder(node->right);
}

/* Driver program to test above functions*/
int main()
{
    struct Node *root = new Node(1);
    root->left = new Node(2);
    root->right = new Node(3);
    root->left->left = new Node(4);
    root->left->right = new Node(5);

    cout << "\nPreorder traversal of binary tree is \n";
    printPreorder(root);

    cout << "\nInorder traversal of binary tree is \n";
    printInorder(root);
}
```

The screenshot shows a Sublime Text editor window with the following details:

- Title Bar:** C:\Users\Ammar\Documents\Naufal Ammar\Naufal Ammar\KULIAH\Smt 2\strukturData\Pert9dan10\3.cpp (KULIAH) - Sublime Text (UN...)
- Menu Bar:** File Edit Selection Find View Goto Tools Project Preferences Help
- Left Panel (File Explorer):** Shows a file tree with several folders and files:
 - Smt 1
 - Smt 2
 - Agama
 - Aljabar Linear
 - Bahasa Indonesia
 - Interaksi Manusia dan I
 - Kebudayaan
 - Organisasi dan Arsitekt
 - Pancasila
 - strukturData
 - Pert2
 - Pert3
 - Pert4
 - Pert5
 - Pert6
 - Pert7
 - Pert8
 - Pert9dan10
 - 1.cpp
 - 2.cpp
 - 3.cpp
 - UAS
 - UTS
 - time.cpp
 - debug.log
 - g.d
 - Logo_Universitas_Singaprel
 - Logo_Universitas_Singaprel
- Center Panel (Code Editor):** Displays the content of the 3.cpp file:

```
/* Given a binary tree, print its nodes in inorder*/
void printInorder(struct Node *node)
{
    if (node == NULL)
        return;

    /* first recur on left child */
    printInorder(node->left);

    /* then print the data of node */
    cout << node->data << " ";

    /* now recur on right child */
    printInorder(node->right);
}

/* Given a binary tree, print its nodes in preorder*/
void printPreorder(struct Node *node)
{
    if (node == NULL)
        return;

    /* first print data of node */
    cout << node->data << " ";

    /* then recur on left subtree */
    printPreorder(node->left);

    /* now recur on right subtree */
    printPreorder(node->right);
}

/* Driver program to test above functions*/
int main()
{
    struct Node *root = new Node(1);
    root->left = new Node(2);
    root->right = new Node(3);
    root->left->left = new Node(4);
    root->left->right = new Node(5);

    cout << "\nPreorder traversal of binary tree is \n";
    printPreorder(root);

    cout << "\nInorder traversal of binary tree is \n";
    printInorder(root);

    cout << "\nPostorder traversal of binary tree is \n";
    printPostorder(root);

    return 0;
}
// GeeksforGeeks
```
- Bottom Status Bar:** Line 12, Column 6, Spaces: 4, C++, 89%, ENG, 15:20

