

# **LAPORAN PRAKTIKUM STRUKTUR DATA**

## **MODUL 10 TREE**



### **Disusun Oleh :**

NAMA : Maulana Ananta Piliang

NIM : 103112400156

### **Dosen**

FAHRUDIN MUKTI WIBOWO

## **PROGRAM STUDI STRUKTUR DATA FAKULTAS INFORMATIKA TELKOM UNIVERSITY PURWOKERTO 2025**

### **A. Dasar Teori**

C++ adalah pengembangan dari bahasa c yang dibuat oleh Bjarne Stroustrup sekitar tahun 1980-an. C++ disebut bahasa multi-paradigma, artinya bisa dipakai dengan gaya prosedural (pakai fungsi biasa), berorientasi objek (pakai class dan object), atau bahkan gabungan keduanya. C++ punya dasar-dasar seperti variabel, operator percabangan (if, switch), perulangan (for, while), dan bisa memakai class untuk membuat objek.

### **B. Guided (berisi screenshot source code & output program disertai penjelasannya)**

Guided 1

H

```
#ifndef BSTREE_H
#define BSTREE_H

typedef int infotype;

struct Node {
    infotype info;
    Node* left;
    Node* right;
};

typedef Node* address;

address alokasi(infotype x); void
insertNode(address &root, infotype x);

void inOrder(address root); void
preOrder(address root); void
postOrder(address root);

int hitungJumlahNode(address root);
int hitungTotal(address root); int
hitungKedalaman(address root, int start);

#endif
```

CPP

```

#include "bstree.h"

#include <iostream> using
namespace std;

address alokasi(infotype x) {
address p = new Node;  p-
>info = x;  p->left =
NULL;  p->right = NULL;
    return p;
}

void insertNode(address &root, infotype x) {
if (root == NULL) {    root = alokasi(x);
} else if (x < root->info) {
insertNode(root->left, x);  } else if (x >
root->info) {    insertNode(root->right,
x);
    }
}

void inOrder(address root) {
if (root != NULL) {

```

```

        inOrder(root->left);
    cout << root->info << " - ";
    inOrder(root->right);
    }
}

void preOrder(address root) {
    if (root != NULL) {        cout
    << root->info << " - ";
    preOrder(root->left);
    preOrder(root->right);
    }
}

void postOrder(address root) {
    if (root != NULL) {
    postOrder(root->left);
    postOrder(root->right);
    cout << root->info << " - ";
    }
}

int hitungJumlahNode(address root) {
    if (root == NULL)
        return 0;
    return 1 + hitungJumlahNode(root->left)
        + hitungJumlahNode(root->right);
}

int hitungTotal(address root) {

```

```

    if (root == NULL)
        return 0;    return root->info +
hitungTotal(root->left)          +
hitungTotal(root->right);
}

int hitungKedalaman(address root, int start) {
    if (root == NULL)
        return start - 1;

    int kiri = hitungKedalaman(root->left, start + 1);
    int kanan = hitungKedalaman(root->right, start + 1);

    return (kiri > kanan) ? kiri : kanan;
}

```

## Main

```

#include <iostream>
#include "bstree.h"

using namespace std;

int main() {    cout << "Hello
World!" << endl;

    address root = NULL;

    insertNode(root, 1);
    insertNode(root, 2);
    insertNode(root, 6);
}

```

```

        insertNode(root, 4);
insertNode(root, 5);
insertNode(root, 3);
insertNode(root, 6);
insertNode(root, 7);

        cout << "InOrder  :
";    inOrder(root);
cout << endl;

        cout << "PreOrder  :
";    preOrder(root);
cout << endl;

        cout << "PostOrder : ";
postOrder(root);    cout
<< endl;

        cout << "Kedalaman : " << hitungKedalaman(root, 0) << endl;
cout << "Jumlah Node : " << hitungJumlahNode(root) << endl;
cout << "Total : " << hitungTotal(root) << endl;

        return 0;
}

```

Screenshots Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

Hello world!
InOrder   : 1 - 2 - 3 - 4 - 5 - 6 - 7 -
PreOrder  : 1 - 2 - 6 - 4 - 3 - 5 - 7 -
PostOrder : 3 - 5 - 4 - 7 - 6 - 2 - 1 -
Kedalaman : 4
Jumlah Node : 7
Total : 28
```

#### Deskripsi:

Program ini merupakan implementasi Binary Search Tree (BST) menggunakan struktur data linked list dalam bahasa C++ BST disusun berdasarkan aturan bahwa nilai pada subtree kiri lebih kecil dari root, sedangkan nilai pada subtree kanan lebih besar dari root. C. Kesimpulan

Kesimpulan dari keempat program C++ tersebut adalah bahwa masing-masing program mengajarkan konsep dasar pemrograman dan logika perhitungan menggunakan array, pointer, fungsi, serta struktur perulangan. Dari program pertama, dipelajari cara mengolah data mahasiswa, menghitung rata-rata, dan menampilkan hasil dalam format tabel. Program kedua memperkenalkan array dinamis dan penggunaan pointer untuk mengakses serta memanipulasi data. Program ketiga menekankan penggunaan fungsi terpisah (modularisasi) untuk menghitung rata-rata, nilai maksimum, dan minimum dari sekumpulan data. Sementara program keempat mengajarkan konsep perulangan bersarang untuk mencetak pola angka berbentuk segitiga. Secara keseluruhan, keempat program ini membantu memahami logika dasar, manipulasi data, penggunaan fungsi, array, pointer, serta struktur kontrol dalam bahasa pemrograman C++.

#### D. Referensi

- Alhazmi, A., & Huang, W. (2020). Teaching loops concept through visualization construction. *International Journal of Instruction*. 13(4), 399-414.
- Sáez-López, J. M., Román-González, M., & Vázquez-Cano, E. (2020). Dampak penggunaan bahasa pemrograman terhadap pemahaman konsep loop pada siswa K-12. *Pendidikan dan Teknologi Informasi*, 25(1), 987-1000.
- Stroustrup, B. (1999). Tinjauan umum bahasa pemrograman C++. *ACM SIGPLAN Notices*, 34(4), 7-18.
- Stroustrup, B. (1999). Tinjauan umum bahasa pemrograman C++. *ACM SIGPLAN Notices*, 34(4), 7-18.