

**LAPORAN PRAKTIKUM
STRUKTUR DATA**

**MODUL 10
PENGENALAN CODE BLOCKS**



Disusun Oleh :

NAMA : Muhamad Ilham Syahid
NIM : 103112400155

Dosen
WAHYU ANDI SAPUTRA

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

A. Dasar Teori

Modul 10 membahas konsep rekursif dan struktur data tree sebagai struktur data non-linear. Rekursif adalah teknik pemrograman di mana fungsi memanggil dirinya sendiri dengan syarat memiliki kondisi berhenti, dan sering digunakan karena penulisan kode lebih sederhana meskipun kurang efisien dalam memori dan waktu. Tree merupakan struktur data hierarkis yang memiliki satu root, node parent-child, serta tidak mengandung siklus. Modul ini menjelaskan terminologi tree, jenis-jenis tree terutama Binary Tree dan Binary Search Tree (BST), serta operasi dasar seperti insert, search, delete, dan traversal (pre-order, in-order, post-order) yang umumnya diimplementasikan secara rekursif.

Guided (berisi screenshot source code & output program disertai penjelasannya)

Guided 1

Bstree.h

```
#ifndef BSTREE_H
#define BSTREE_H

#define Nil NULL
typedef int infotype;

typedef struct Node* address;

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

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

#endif
```

Bstree.cpp

```
#ifndef BSTREE_H
#define BSTREE_H

#define Nil NULL
typedef int infotype;

typedef struct Node* address;
```

```
struct Node {
    infotype info;
    address left;
    address right;
};

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

#endif
```

Main.cpp

```
#include <iostream>
#include "bstree.h"
using namespace std;

int main() {
    address root = Nil;

    insertNode(root,1);
    insertNode(root,2);
    insertNode(root,6);
    insertNode(root,4);
    insertNode(root,5);
    insertNode(root,3);
    insertNode(root,7);

    printInorder(root);
    return 0;
}
```

Screenshots Output

```
PS C:\Pratikum Struktur Data\modul 10 laprak> g++ main.cpp bstree.cpp -o main
PS C:\Pratikum Struktur Data\modul 10 laprak> ./main
1 2 3 4 5 6 7
PS C:\Pratikum Struktur Data\modul 10 laprak> []
```

Deskripsi:

Program ini mengimplementasikan Binary Search Tree (BST) menggunakan linked list dan konsep rekursif. File bstree.h berisi deklarasi struktur node dan fungsi, bstree.cpp berisi implementasi fungsi seperti alokasi node, insert data sesuai aturan BST, pencarian data, dan traversal inorder, sedangkan main.cpp digunakan untuk menguji program dengan memasukkan data dan menampilkan hasilnya. Selain itu, program menyediakan fungsi rekursif untuk menghitung jumlah node, total nilai data, dan kedalaman tree, serta traversal pre-order, in-order, dan post-order.

Kesimpulan

Kesimpulan dari codingan di atas adalah bahwa program berhasil mengimplementasikan struktur data Binary Search Tree (BST) menggunakan linked list dan fungsi rekursif. Program mampu melakukan operasi dasar BST seperti penyisipan data, pencarian, dan traversal, serta menghitung jumlah node, total nilai, dan kedalaman tree. Hasil traversal inorder menampilkan data secara terurut, sesuai dengan karakteristik Binary Search Tree, sehingga program efektif untuk menyimpan dan mengelola data secara hierarkis.

B. Referensi

Struktur Data – Modul 10: Tree (Bagian Pertama).