****

**LAPORAN PRAKTIKUM**

* **Identitas Praktikum**

Nama MK : Struktur Data

Kode MK : CCK2AAB4

Bobot SKS : 4 SKS

Tempat : L-Program, Gedung DC, lantai 3

Hari, tanggal : Selasa, 26 November 2024

Jam : 12:30-15:30 WIB

Topik praktikum : Modul-10 TREE

* **Identitas Mahasiswa**

Nama lengkap : Afad Fath Musyarof Halim

NIM : 2211104030

Program Studi : S-1 Software Engineering

* **Hasil Praktikum**

1. TREE
   1. Recursive

Adalah fungsi yang memanggil dirinya sendiri

Contoh:

* Recursive menghitung Faktorial
  + Code  
    A screen shot of a computer program

    Description automatically generated
  + Output  
    A screen shot of a computer program

    Description automatically generated
* Recursive menampilkan angka berurutan
  + Code  
    A screen shot of a computer code

    Description automatically generated
  + Output  
    A screen shot of a computer program

    Description automatically generated
  1. Ordered Tree

Adalah metode struktur data dimana data tidak memiliki urutan tertentu namun merupakan turunan dari suatu angka untuk setiap data yang di inputkan kecuali urutan paling awal (root)

Contoh:

* Code  
  A screen shot of a computer code

  Description automatically generated
* Output  
  A screen shot of a computer

  Description automatically generated
  1. Binary Tree

Adalah Tree yang hanya memiliki maksimal 2 turunan.

* Koding  
  A screenshot of a computer program

  Description automatically generated
* Output  
  A screenshot of a computer

  Description automatically generated
  1. Traversal Binary Tree

Traversal adalah metode untuk mencetak nilai dari setiap elemen yang ada pada tree

Contoh apabila menggunakan kodingan binary tree di atas (No 10.3)

* Kodingan  
  A computer screen shot of text

  Description automatically generated
* Output  
  A number on a black background

  Description automatically generated
  1. Soal
     + 1. Membuat ADT Binary Tree
* Coding
  + Bstree.h  
    A screen shot of a computer program

    Description automatically generated
  + Bstree.cpp  
    A screenshot of a computer program

    Description automatically generated
  + Main.cpp  
    A screen shot of a computer program

    Description automatically generated
* Output

A black background with white text

Description automatically generated

* Penjelasan
  + *Alokasi* untuk membuat Node baru sebelum dimasukan ke tree
  + *Insert Node* untuk memasukan Node yang telah dibuat ke posisi berdasarkan prinsip binary tree yang akan di tempatkan sebagai turunan (child) dari node yang sesuai, jika lebih kecil dari node akan di sebelah kiri, jika lebih besar akan di sebelah kanan. Jika nilai node sudah ada pada tree maka tidak akan dimasukan
  + *Find node* untuk mencari node berdasarkan inputan lalu di iterasikan posisinya seperti saat insert namun daripada menambahkan node, fungsi akan mengembalikan nilai dari inputan apabila inputan ditemukan pada tree
  + *Print inorder* untuk menampilkan seluruh elemen pada tree dari yang paling kiri(kecil) ke paling kanan (besar)
    - 1. Buat Fungsi tambahan
* fungsi hitungJumlahNode
* fungsi hitungTotalInfo
* fungsi hitungKedalaman
  + tambahan Coding
    - bstree.h  
      A screen shot of a computer program

      Description automatically generated
    - bstree.cpp  
      A screen shot of a computer program

      Description automatically generated
    - main.cpp  
      A screenshot of a computer code

      Description automatically generated
  + Output  
    A screenshot of a computer program

    Description automatically generated
  + Penjelasan
    - *hitungJumlahNode* berfungsi dengan menambahkan nilai 1 untuk setiap node yang ada, jika node sudah tidak memiliki turunan maka fungsi akan menghentikan perhitungan
    - *hitungTotalInfo* berfungsi dengan menjumlahkan setiap nilai dari masing-masing node
    - *hitungKedalaman* berfungsi dengan menelusuri turunan dari root bagian kiri dan kanan lalu menambahkan nilai turunan 1 untuk setiap 1 kali turun, lalu dari turunan kiri dan kanan dibandingkan dan mengembalikan nilai dengan nilai turunan terbanyak