

# LEMBAR KERJA PRAKTIKUM ALGORITMA DAN STRUKTUR DATA

Dosen Pengampu:

1. Nurul Fathanah Mustamin, S.Pd., M.T
2. Andry Fajar Zulkarnain, S.ST., M.T

## Modul 7. Tree (Pohon)

### Tujuan Praktikum:

1. Mahasiswa mampu menjelaskan mengenai konsep Tree.
2. Mahasiswa mampu membuat dan mendeklarasikan struktur algoritma Tree
3. Mahasiswa mampu menerapkan dan mengimplementasikan algoritma Tree

### 1. Soal Praktikum:

Cobalah program berikut, perbaiki output, lengkapi fungsi inOrder dan postOrder pada coding, running, simpan program !

```
1  #include <stdio.h>
2  #include <conio.h>
3  #include <stdlib.h>
4  #include <iostream>
5
6  using namespace std;
7  struct Node
8  {
9      int data;
10     Node *kiri;
11     Node *kanan;
12 };
13
14 void tambah(Node **root, int databaru)
15 {
16     if (*root == NULL)
17     {
18         Node *baru;
19         baru = new Node;
20         baru->data = databaru;
21         baru->kiri = NULL;
22         baru->kanan = NULL;
23         (*root) = baru;
24         (*root)->kiri = NULL;
25         (*root)->kanan = NULL;
26         cout << "Data bertambah";
27     }
28     else if (databaru < (*root)->data)
29         tambah(&(*root)->kiri, databaru);
30     else if (databaru > (*root)->data)
31         tambah(&(*root)->kanan, databaru);
32     else if (databaru == (*root)->data)
33         cout << "Data sudah ada";
34 }
35
36 void preOrder(Node *root)
37 {
38     if (root != NULL)
39     {
40         cout << root->data;
41         preOrder(root->kiri);
42         preOrder(root->kanan);
43     }
44 }
45
46 void inOrder(Node *root)
47 {
48     if (root != NULL)
```

```

49
50
51
52
53
54 }
55
56 void postOrder(Node *root)
57 {
58
59
60
61
62
63
64 }
65
66 int main()
67 {
68     int pil, data;
69     Node *pohon;
70     pohon = NULL;
71     do
72     {
73         system("cls");
74         cout << "1. Tambah\n";
75         cout << "2. PreOrder\n";
76         cout << "3. inOrder\n";
77         cout << "4. PostOrder\n";
78         cout << "5. Exit\n";
79         cout << "\nPilihan : ";
80         cin >> pil;
81         switch (pil)
82         {
83             case 1:
84                 cout << "\n INPUT : ";
85                 cout << "\n -----";
86                 cout << "\n Data baru : ";
87                 cin >> data;
88                 tambah(&pohon, data);
89                 break;
90             case 2:
91                 cout << "PreOrder";
92                 cout << "\n-----\n";
93                 if (pohon != NULL)
94                 {
95                     preOrder(pohon);
96

```

```

97         else
98         |     cout << "Masih Kosong";
99         |     break;
100     case 3:
101         |     cout << "InOrder";
102         |     cout << "\n-----\n";
103         |     if (pohon != NULL)
104         |     {
105         |         |     inOrder(pohon);
106         |         |     }
107         |     else
108         |         cout << "Masih Kosong";
109         |     break;
110     case 4:
111         |     cout << "PostOrder";
112         |     cout << "\n-----\n";
113         |     if (pohon != NULL)
114         |     {
115         |         |     postOrder(pohon);
116         |         |     }
117         |     else
118         |         cout << "Masih Kosong";
119         |     break;
120     case 5:
121         |     return 0;
122     }
123     _getch();
124 } while (pil != 5);
125 return EXIT_FAILURE;
126 }

```