

# MODUL PRAKTIKUM STRUKTUR DATA “PREORDER, INORDER DAN POSTORDER”



**Matheus S. Rumetna, S.Kom**

(email : [matheus.rumetna@gmail.com](mailto:matheus.rumetna@gmail.com))

## MODUL 6

### PREORDER, INORDER DAN POSTORDER

#### A. TUJUAN

1. Mengetahui apa yang dimaksud dengan preorder, inorder, dan postorder
2. Memahami cara kerja kunjungan preorder, inorder, dan postorder pada pohon

#### B. PERALATAN

1. Personal komputer (PC)
2. Kertas Kosong
3. Alat tulis menulis
4. Disket kerja
5. Buku struktur data

#### C. LANDASAN TEORI

Atas sebuah pohon kita dapat melakukan sejumlah operasi, salah satu operasi yang sering dilakukan adalah melakukan kunjungan pada setiap simpul pada suatu pohon biner tepat satu kali dengan melakukan kunjungan secara lengkap. Seperti yang telah dijelaskan di atas bahwa urutan informasi yang tersimpan dalam pohon biner akan berbeda jika letak simpul ditukar. Dengan alasan inilah kita bisa melakukan kunjungan dengan tiga cara yaitu secara preorder, inorder, dan postorder.

Kunjungan preorder dilaksanakan dengan mencetak simpul yang dikunjungi, kunjungan cabang kiri dan kunjungan cabang kanan.

Kunjungan inorder dimulai dari kunjungan ke cabang kiri, kemudian mencetak akar dan diteruskan dengan kunjungan ke cabang kanan. Ketiga macam kunjungan ini akan kita perlakukan kembali terhadap cabang kiri atau cabang kanan.

Uru-urutan dalam kunjungan postorder adalah kunjungi cabang kiri, cabang kanan dan cetak simpul. Langkah ini diulang untuk cabang kiri dan cabang kanan.

#### D. TUGAS PENDAHULUAN

1. Buatlah algoritma dari cuplikan program dibawah ini
2. Ketiklah program dibawah ini.

```
uses crt;
const
    max=10;
type
    arr=array[1..max] of byte;
var
    i      :byte;
    data   :arr;
procedure input;
begin
    clrscr;
    writeln(' Insert 10 Bilangan : ');
    writeln('=====');
    for i:=1 to max do
    begin
        write('Data ke-',i,'=');readln(data[i]);
    end;
    clrscr;
    for i:=1 to max do
        write(data[i] , ' ');
    writeln;
    writeln('Data Yang Terurut :');
    writeln('=====');
```

```

end;

procedure Move(x,y:byte); {untuk menggeser data ke depana
satu
var i:byte;                langkah }
begin
  for i:= y downto x do
    data[i]:=data[i-1];
  end;

procedure FindPos(x:byte; var y:byte);

var i:byte;
begin
  y:=x; i:=0;
  repeat
    inc (i)
  until data[x]<=data[i];
  y:=i;
end;
procedure insert;
var
  i,j,temp :byte;
begin
  for i:=2 to max do
    begin
      temp:= data[i]; findpos(i,j);
      move(j,i);data[j]:=temp;
      end;
    end;

procedure cari;
function search(x,n:integer):integer;
var
  find      : boolean;
  l,r,mid,temp: integer;
begin
  l:=1; r:=n;
  find:=false;
  repeat
    mid:=(l + r) div 2;
    if x=data[mid] then
      begin
        temp:=mid;find:=true;
      end else
        if x < data[mid] then
          r:=mid-1
        else
          l:=mid+1;
        until find or (l > r);
        if not find then temp:=0;
        search:=temp;
      end;
var bil:byte;
begin
  gotoxy(1,7); write(' Bilangan yang dicari :
');readln(bil);
  gotoxy(1,8); write(' Data yang dicari pada urutan ke- ',
search(bil,max));
end;

```

```

procedure output;
begin
  for i := 1 to max do
    write(data[i] , ' ');
  end;
begin
input;
insert;
output;
cari;readkey;
end.

```

```

Insert 10 Bilangan :
=====
Data ke-1=5
Data ke-2=2
Data ke-3=67
Data ke-4=43
Data ke-5=98
Data ke-6=45
Data ke-7=8
Data ke-8=23
Data ke-9=39
Data ke-10=?

```

```

5 2 67 43 98 45 8 23 39 ?
Data Yang Terurut =
=====
2 5 7 5 23 39 43 45 67 98
Bilangan yang dicari = 43
Data yang dicari pada urutan ke- ?

```

Matheus Rumetna

```

Uses crt;

Const
    Max = 5;

Type
Tipe_data = byte;
Rec_arr = array[1..max] of tipe_data;

Var
    I : byte;
    Data :rec_arr;

Procedure CetakLangkah (data:rec_arr;step,source,target:byte);
Var I : byte;
Begin
    Gotoxy (20,7+step);write('I   I   I   I   I   I');
    Gotoxy (4,7+step);write('Langka ', step,':');
    For I := 1 to max do Begin
        Gotoxy (17+i*5,7+step);write(data[i]);
    End;
    Gotoxy (47,7+step);write('data ',source,' Insert data ',target);
End;

Procedure insert (var data:rec_arr;mulai,akhir:byte);
Var temp, I : byte;
Begin
    Temp := data[akhir];
    For I := akhir downto mulai do
        Begin
            Data[i] := data[i-1];
        End;
    Data[mulai]:= temp;
End;

Procedure insertanim (var data:rec-arr; jml_data:byte);
Var
    Ketemu :Boolean;

```

```

Count,I,x,step :byte;

Begin

Step := 0;

For I := 2 to jml_data do

Begin

Ketemu := false;

Count := 0;

Repeat

Inc (count);

If data[i] < data[count] then

Begin

Inc (step);

Insert (data,count,i);

Ketemu := true;

End;

Until ketemu of (count = i);

If ketemu then cetaklangkah (data,step,I,count);

End;

Gotoxy(67,7+step);write('→ tersort ');

Gotoxy(20,8+step);write ('I__I__I__I__I__I ');

End;

Procedure cetakdata;

Var I : byte;

Begin

Clrscr;

Gotoxy(30,1);write('Insertion Sort');

Gotoxy(30,1);write('=====');

Gotoxy(30,1);write('');
-----

Gotoxy(30,1);write('');
III I IV I ');

Gotoxy(30,1);write('');
I-----I-----I-----

I-----I ');

```



## LEMBAR KONTROL

N A M A : \_\_\_\_\_

N P M : \_\_\_\_\_

KELAS : \_\_\_\_\_

SEMESTER : \_\_\_\_\_

No.	Materi Praktikum	Hari/tanggal	Paraf Dosen
1			
2			
3			
4			
5			
6			
7			
8			