A logo of a university

Description automatically generated

NAMA : RAKASYAEL HIZKIA KOLONDAM

NIM : 2023105490

MATA KULIAH : TI0091 – PRAKTIKUM DATA TERSTRUKTUR

PRAKTIKUM KE - : 11

|  |  |  |
| --- | --- | --- |
| **No.** | **Kode** | **Penjelasan** |
| 1 | static int factorial(int n)  {  if(n == 1)  {  return 1;  }  else {  return n \* factorial(n-1);  }  } | Metode ini mengembalikan nilai faktorial n. Cara kerja dari metode ini adalah melakukan pengecekan terlebih dahulu terhadap input. Lalu jika input bukan sama dengan 1 maka akan mengembalikan nilai n dikalikan n-1. Pada pengembalian n-1 akan terjadi lagi hal yang sama sampai nilai n = 1. Jika sudah sama dengan 1, maka sebua hasil rekursi yang mengembalikan nilai n, n-1, n-2, …, 1 akan dikalikan semua sehingga menghasilkan nilai n! |
| 2 | public void **inOrder**(Node node)    {  *//Modifikasi bagian ini*  *if*(node **!=** null){  **inOrder**(node.left);        System.out.**print**(node.value **+** " ");  **inOrder**(node.right);      }    } | Metode dalam BST ini akan menampilkan setiap node menggunakan cara in order transversal. Metode ini bekerja dengan cara mengecek node terlebih dahulu, jika terdapat node di dalam tree, maka akan melakukan pencarian ke paling kiri bawah terlebih dahulu, lalu menampilkannya, lalu lanjut ke kanan. |
| 3 | public void **preOrder**(Node node)    {  *//Modifikasi bagian ini*  *if*(node **!=** null){        System.out.**print**(node.value **+** " ");  **preOrder**(node.left);  **preOrder**(node.right);      }    } | Metode dalam BST ini menampilkan setiap node di dalam tree dengan metode pre order transversal. Dimana dilakukan terlebih dahulu pengecekan lalu jika node tidak null maka akan menampilkan root node terlebih dahulu lalu satu persatu pergi ke kiri, lalu ke kanan, lalu menampilkannya lagi. |
| 4 | public void **postOrder**(Node node)    {  *//Modifikasi bagian ini*  *if*(node **!=** null){  **postOrder**(node.left);  **postOrder**(node.right);        System.out.**print**(node.value **+** " ");      }      } | Sama seperti pre order transversal, Post order transversal merupakan metode untuk menampilkan setiap node yang ada di dalam tree. Hal ini dilakukan dengan cara pengecekan terlebih dahulu node, jika tidak null maka akan melakukan rekursi ke kiri terlebih dahulu lalu mengecek yang di kanan, lalu menampilkannya. Hal ini dilakukan hingga semua node di visit. |