A logo of a university

Description automatically generated

NAMA : RAKASYAEL HIZKIA KOLONDAM

NIM : 2023105490

MATA KULIAH : TI0091 – PRAKTIKUM DATA TERSTRUKTUR

PRAKTIKUM KE - : 4

|  |  |  |
| --- | --- | --- |
| **No.** | **Kode** | **Penjelasan** |
| 1 | public Stack()  {  top = null;  height = 0;  } | Ini adalah konstruktor dari kelas Stack untuk assign value pada top yaitu null dan tinggi stack 0. Value ini akan berubah seiring perubahan dari stack. |
| 2 | public Stack(int value)  {  Node newNode = new Node(value);  top = newNode;  height = 1;  } | Ini adalah kelas untuk memanggil stack baru yang akan digunakan serta mengisi stack dengan 1 node. |
| 3 | public void printStack()  {  Node currentNode = top;  while(currentNode != null)  {  System.out.print(currentNode.value + " ");  currentNode = currentNode.next;  }  System.out.println();  } | Ini adalah kelas untuk menampilkan isi dari stack dengan cara membaca value dari setiap node satu persatu dari top sampai ke node terakhir. |
| 4 | public void push(int *value*) {  Node newNode = new Node(*value*);  if(top == null){  top = newNode;  }  else{  newNode.next = top;  top = newNode;  }  height++; } | Ini adalah metode push dari stack untuk memasukkan node ke dalam stack. Akan dibuat node baru sesuai inputtan lalu akan melakukan pengecekan jika stack kosong maka akan newnode akan ditempatkan di top atau ke bagian paling atas dari stack. Atau jika tidak maka newnode akan menjadi top dengan menunjuk node selanjutnya yang awalnya adalah top menjadi node setelah newNode. Lalu setelah itu menambah tinggi dari stack. |
| 5 | public int pop() {  if(top == null){  System.*err*.println("Empty list");  return -1;  }  else{  Node current = top;  top = top.next;  current.next = null;  height--;  return current.value;  } } | Ini adalah fungsi pop yang mengembalikan integer. Pada fungsi ini akan dilakukan pengecekan terlebih dahulu, jika stack tidak kosong akan dilakukan penghapusan node dengan cara mengubah node setelah top menjadi node top yang baru dan menghapus top (sekarang current) lalu mengembalikan value dari top yang dihapus. |
| 6 | public static void decimalToBinary(int *decimal*) {  Stack stack = new Stack();   if(*decimal* == 0){  int x = 0;  System.*out*.println(x);  }   int n = *decimal*;  while(n > 0){  int rem = n % 2;  stack.push(rem);  n /= 2;  }   stack.printStack(); } | Ini adalah metode untuk mengubah bilangan desimal menjadi bilangan biner. Pertama dilakukan pemanggilan pada kelas stack untuk membuat stack yang baru, lalu akan dilakukan pengulangan dengan assign value n menjadi bilangan inputtan. Selama n lebih dari 0 atau bilangannya belum habis, maka hasil bagi dari bilangan akan di push ke dalam stack lalu membagi bilangan dengan 2 sehingga habis. Setelah itu hasil nya akan di display. |