**Worksheet pertemuan 6 - 2**

**Algoritma dan Struktur Data**

**LinkedList**

**NIM : 23523170**

**Nama : Danendra Farrel Adriansyah**

1. **Membuat Folder Untuk Menyimpan Hasil Praktikum**
2. Siapkan folder kosong dengan nama menggunakan NIM masing-masing. Jika folder NIM pada pertemuan sebelumnya mau dimanfaatkan, jangan lupa pindahkan dulu isinya ke folder lain sebagai arsip.
3. Folder ini akan dijadikan tempat untuk menyimpan semua pdf dari worksheet ini beserta file praktikum lainnya.
4. **Membuat class LinkedList**
   1. Buatlah projek dengan nama **TestLinkedList**
   2. Pada projek tersebut, silakan buat sebuah class dengan nama **LinkedList**
   3. Kemudian salin tempel kode pogram di bawah ini

|  |
| --- |
| package testlinkedlist;  /\*\*  \*  \* @param <E>  \*/  public class LinkedList<E> {  Element first;  Element last;  int size = 0;    private static final class Element<E>{  E data;  Element next;  Element previous;    private Element(E data){  this.data = data;  }  }    public LinkedList(){    }    public Object getFirst(){  if(size == 0){  return null;  }else{  return first.data;  }  }    public Object getLast(){  if(size == 0){  return null;  }else{  return last.data;  }  }    public Object get(int index){  if (index < 0 || index >= size){  System.out.println("Index out of bounds");  System.exit(-1);  }  return getElement(index).data;  }    Element getElement(int n){  Element e;  if (n < size / 2){  e = first;  // n less than size/2, iterate from start  while (n-- > 0){  e = e.next;  }  }else{  e = last;  // n greater than size/2, iterate from end  while (++n < size){  e = e.previous;  }  }  return e;  }    void removeElement(Element e){  size--;  if (size == 0){  first = last = null;  }else{  if (e == first){  first = e.next;  e.next.previous = null;  }else if(e == last){  last = e.previous;  e.previous.next = null;  }else{  e.next.previous = e.previous;  e.previous.next = e.next;  }  }  }    public boolean remove(Object o){  Element e = first;    while (e != null){  if (o.equals(e.data)){  removeElement(e);  return true;  }  e = e.next;  }  return false;  }    public Object remove(int index){  if (index < 0 || index > size){  System.out.println("Index out of bounds");  System.exit(-1);  }  Element e = getElement(index);  removeElement(e);  return e.data;  }    public Object remove(){  return removeFirst();  }    public Object removeFirst(){  if(size == 0){  return null;  }else{  size--;  Object removed = first.data;    if (first.next != null){  first.next.previous = null;  }else{  last = null;  }  first = first.next;  return removed;  }  }    public Object removeLast(){  if(size == 0){  return null;  }else{  size--;  Object removed = last.data;    if (last.previous != null){  last.previous.next = null;  }else{  first = null;  }  last = last.previous;  return removed;  }  }    public void add(E obj){  addLast(obj);  }    public void addFirst(E obj){  Element e = new Element(obj);    if (size == 0){  first = last = e;  }else{  e.next = first;  first.previous = e;  first = e;  }  size++;  }    public void addLast(E obj){  addLastElement(new Element (obj));  }    private void addLastElement(Element e){  if (size == 0){  first = last = e;  }else{  e.previous = last;  last.next = e;  last = e;  }  size++;  }    public void add(int index, E obj){  if (index < 0 || index > size){  System.out.println("Index out of bounds");  System.exit(-1);  }  Element e = new Element(obj);  if (index < size){  Element after = getElement(index);  e.next = after;  e.previous = after.previous;  if (after.previous == null){  first = e;  }else{  after.previous.next = e;  }  after.previous = e;  size++;  }else{  addLastElement(e);  }  }    public void clear(){  if (size > 0){  first = null;  last = null;  size = 0;  }  }    public int size(){  return size;  }    public boolean contains(Object o){  Element e = first;  while (e != null){  if (o.equals(e.data)){  return true;  }  e = e.next;  }  return false;  }  } |

1. **Membuat dan Menjalankan Main Method ke-1**
   1. Bukalah class **TestLinkedList** yang masih kosong
   2. Kemudian salin tempel kode program di bawah ini (timpa ke dalam template), ada error-nya kan, coba ikuti pesan error itu untuk membetulkannya sehingga main method ini dapat di-run.

|  |
| --- |
| package testlinkedlist;  public class TestLinkedList {  public static void main(String[] args) {  LinkedList<String> cars = new LinkedList<>();    cars.add("Volvo");  cars.add("BMW");  cars.add("Ford");  cars.add("Mazda");    cars.addFirst("Toyota");  cars.addLast("Mitsubishi");    System.out.println("Size of Linked List: "+cars.size());  System.out.println("First element: "+cars.getFirst());  System.out.println("Last element: "+cars.getLast());  System.out.println("Element at index 4 -> "+cars.get(4));    cars.add(6, "Wuling");  System.out.println("\nAdd Wuling at index 6");  System.out.println("Wuling in this LinkedList? "+cars.contains("Wuling"));    System.out.println("Remove first -> "+cars.removeFirst());    System.out.println("Remove last -> "+cars.removeLast());    System.out.println("Remove at index 3 -> "+cars.remove(3));  System.out.println("\nSize of Linked List: "+cars.size());    System.out.println("\nAdd last -> Nissan");  cars.addLast("Nissan");  System.out.println("Add first -> Ferrari");  cars.addFirst("Ferrari");  System.out.println("\nSize of Linked List: "+cars.size());  System.out.println("First element: "+cars.getFirst());  System.out.println("Last element: "+cars.getLast());    /\*  Replace this code to display all elements of LinkedList  \*/  }  } |

* 1. Jalankan *main method* **TestLinkedList** dan hasil tangkapan layar keluaran dari program silakan letakkan di bawah ini

|  |
| --- |
|  |

1. **Membuat Class Mahasiswa**
   1. Anda akan disuruh membuat class dengan nama **Mahasiswa**
   2. Class ini nanti akan menjadi tipe data dari LinkedList yang akan dibuat (pada praktik poin C menggunakan tipe data String)
   3. Salin tempel kode program di bawah ini di class **Mahasiswa** yang Anda buat

|  |
| --- |
| package testlinkedlist;  public class Mahasiswa {  String nama = null;  String NIM = null;  Double IPK = 0.0;    public Mahasiswa(String NIM, String nama, Double ipk){  this.NIM = NIM;  this.nama = nama;  this.IPK = ipk;  }    public String getNama(){  return this.nama;  }    public String getNIM(){  return this.NIM;  }    public Double getIPK(){  return this.IPK;  }    /\*  Ganti kode ini untuk menambahkan method setNama(), setNIM() dan setIPK()  \*/  } |

* 1. Class **Mahasiswa** mempunyai 3 atribut yaitu nama, NIM dan IPK
  2. *Method* yang dimiliki yaitu
     + getNama() – untuk mengambil data nama dari objek mahasiswa
     + getNIM() - untuk mengambil data NIM dari objek mahasiswa
     + getIPK() - untuk mengambil data IPK dari objek mahasiswa

1. **Membuat dan Menjalankan Main Method ke-2**
   * + 1. Buatlah *main method* dengan nama **Main** salin tempel kode program di bawah ini

|  |
| --- |
| package testlinkedlist;  public class Main {  public static void main(String[] args) {    LinkedList<Mahasiswa> mhs = new LinkedList<>();    mhs.add(new Mahasiswa("16523001","Idris",3.88));  mhs.add(new Mahasiswa("16523002","Agus",3.18));  mhs.add(new Mahasiswa("16523003","Dani",3.42));  mhs.add(new Mahasiswa("16523008","Roni",3.67));  mhs.add(new Mahasiswa("16523020","Sinta",3.05));    LinkedList.Element ptr = mhs.first;    while(ptr != null) {  Mahasiswa m = (Mahasiswa) ptr.data;  System.out.println("NIM: "+m.getNIM());  System.out.println("Nama: "+m.getNama());  System.out.println("IPK: "+m.getIPK());  System.out.println("");  ptr = ptr.next;  }    /\*  Ganti kode ini untuk mengganti IPK mahasiswa di indeks ke-3  Kemudian tampilkan data mahasiswa yang ada di indeks ke-3  \*/  }  } |

Jalankan *main class* **Main** (yang baru saja dibuat) dan taruh hasil tangkapan layar dari keluarannya di bawah ini

|  |
| --- |
|  |

* + - 1. Pada class **Mahasiswa** silakan tambahkan beberapa *method* di bawah ini dan taruh kode programnya di kotak di bawah ini juga

**setNama() – untuk mengeset/mengganti nama mahasiswa**

|  |
| --- |
| public void setNama(String *nama*) {  *this*.nama = *nama*;      } |

**setNIM() – untuk mengeset/mengganti NIM mahasiswa**

|  |
| --- |
| public void setNIM(String *NIM*) {  *this*.NIM = *NIM*;      } |

**setIPK() – untuk mengeset/mengganti IPK mahasiswa**

|  |
| --- |
| public void setIPK(Double *IPK*) {  *this*.IPK = *IPK*;      } |

* + - 1. Pada bagian akhir main class Main silakan tambahkan beberapa kode untuk mengganti nama, NIM dan IPK mahasiswa yang berada di indeks ke-3 (menggunakan method **setNama()**, **setNIM()** dan **setIPK()** yang sudah Anda buat) dan menampilkan data mahasiswa tersebut saja. Tulis kode programnya di bawah ini juga.

|  |
| --- |
| public class Main {      public static void main(String[] *args*) {          LinkedList<Mahasiswa> mhs = new LinkedList<>();          mhs.add(new Mahasiswa("16523001", "Idris", 3.88));          mhs.add(new Mahasiswa("16523002", "Agus", 3.18));          mhs.add(new Mahasiswa("16523003", "Dani", 3.42));          mhs.add(new Mahasiswa("16523008", "Roni", 3.67));          mhs.add(new Mahasiswa("16523020", "Sinta", 3.05));          LinkedList.Element counter = mhs.first;          int count = 0;          while (counter != null && count < 3) {              counter = counter.next;              count++;          }          if (counter != null) {              Mahasiswa m = (Mahasiswa) counter.data;              m.setIPK(3.8);              System.out.println("NIM  : " + m.getNIM());              System.out.println("Nama : " + m.getNama());              System.out.println("IPK  : " + m.getIPK());          }        }  } |

* + - 1. Jalankan kembali *main class* **Main** dan taruh hasil tangkapan layar dari keluarannya di bawah ini

|  |
| --- |
|  |

**\*Catatan**

* **Jangan lupa simpan juga file worksheet ini (yang sudah diisi) sebagai file pdf di folder NIM anda.**
* **Pastikan file TestLinkedList.java, Mahasiswa.java** dan **Main.java sudah tersimpan di dalam folder yang Anda gunakan**
* **Kompres folder ini sebagai file ZIP kemudian kumpulkan di classroom atau ruang pengumpulan lain di kelas masing-masing.**