package zad1;  
  
public class Main {  
 public static void main(String[] args){  
 CycledLinkedList<Integer> list = new CycledLinkedList<>();  
  
 System.*out*.println("Size: " + list.size());  
 list.addEnd(15);  
 System.*out*.println("List contains 15: " + list.contains(15));  
 System.*out*.println("List contains 25: " + list.contains(25));  
 list.addEnd(25);  
 list.addEnd(35);  
 list.addEnd(55);  
 list.addEnd(65);  
 list.addEnd(75);  
 System.*out*.println("Size: " + list.size());  
 list.printList();  
  
 list.insert(3, 15093);  
 list.printList();  
 list.insert(8, -150);  
 list.printList();  
  
 System.*out*.println("Deleting at index 1: " + list.deleteIndex(1).getValue());  
 System.*out*.println(list.get(3));  
 System.*out*.println(list.get(9));  
 System.*out*.println("Deleting at index 0: " + list.deleteIndex(0).getValue());  
  
 list.set(5, 55555);  
 list.set(12, 2251);  
 list.printList();  
  
 System.*out*.println("List contains 25: " + list.contains(25));  
 System.*out*.println("List contains 27: " + list.contains(27));  
  
 System.*out*.println(list.deleteIndex(3).getValue());  
 list.printList();  
 System.*out*.println(list.delete(15093));  
 list.printList();  
 System.*out*.println(list.deleteEl(new Element(65)));  
 list.printList();  
 list.clear();  
 list.printList();  
 }  
}

package zad1;  
  
  
public class Element<T> {  
 private T value;  
 private Element next;  
  
 public Element(T value) { this.value = value; }  
  
 public T getValue() { return value; }  
  
 public void setValue(T val) { value = val; }  
  
 public Element getNext() { return next; }  
  
 public void setNext(Element element) { next = element; }  
  
 @Override  
 public boolean equals(Object obj) {  
 if(obj == null) return false;  
  
 if(obj.getClass() != this.getClass()) return false;  
  
 Element e = (Element)obj;  
  
 return this.getValue().equals(e.getValue());  
 }  
}

package zad1;  
  
public interface IList<E> {  
 void addEnd(E e);  
 void insert(int pos, E e);  
 E get(int pos);  
 void set(int pos, E e);  
 boolean contains(E e);  
 int size();  
 void clear();  
 Element deleteIndex(int pos);  
 boolean delete(E e);  
 E deleteEl(Element e);  
}

package zad1;  
  
public class Main {  
 public static void main(String[] args){  
 CycledLinkedList<Integer> list = new CycledLinkedList<>();  
  
 System.*out*.println("Size: " + list.size());  
 list.addEnd(15);  
 System.*out*.println("List contains 15: " + list.contains(15));  
 System.*out*.println("List contains 25: " + list.contains(25));  
 list.addEnd(25);  
 list.addEnd(35);  
 list.addEnd(55);  
 list.addEnd(65);  
 list.addEnd(75);  
 System.*out*.println("Size: " + list.size());  
 list.printList();  
  
 list.insert(3, 15093);  
 list.printList();  
 list.insert(8, -150);  
 list.printList();  
  
 System.*out*.println("Deleting at index 1: " + list.deleteIndex(1).getValue());  
 System.*out*.println(list.get(3));  
 System.*out*.println(list.get(9));  
 System.*out*.println("Deleting at index 0: " + list.deleteIndex(0).getValue());  
  
 list.set(5, 55555);  
 list.set(12, 2251);  
 list.printList();  
  
 System.*out*.println("List contains 25: " + list.contains(25));  
 System.*out*.println("List contains 27: " + list.contains(27));  
  
 System.*out*.println(list.deleteIndex(3).getValue());  
 list.printList();  
 System.*out*.println(list.delete(15093));  
 list.printList();  
 System.*out*.println(list.deleteEl(new Element(65)));  
 list.printList();  
 list.clear();  
 list.printList();  
 }  
}

Wyniki działania programu:

Size: 0

List contains 15: true

List contains 25: false

Size: 6

[15, 25, 35, 55, 65, 75]

[15, 25, 35, 15093, 55, 65, 75]

[15, -150, 25, 35, 15093, 55, 65, 75]

Deleting at index 1: -150

15093

35

Deleting at index 0: 15

[2251, 35, 15093, 55, 65, 55555]

List contains 25: false

List contains 27: false

55

[2251, 35, 15093, 65, 55555]

true

[2251, 35, 65, 55555]

65

[2251, 35, 55555]

[]