class Node {

int data;

Node next;

public Node(int data) {

this.data = data;

this.next = null;

}

}

public class CircularLinkedList {

private Node head;

public CircularLinkedList() {

this.head = null;

}

public void insert(int data) {

Node newNode = new Node(data);

if (head == null) {

head = newNode;

newNode.next = head;

} else if (data <= head.data) {

Node last = getLastNode();

last.next = newNode;

newNode.next = head;

head = newNode;

} else {

Node current = head;

while (current.next != head && data > current.next.data) {

current = current.next;

}

newNode.next = current.next;

current.next = newNode;

}

}

public Node getLastNode() {

if (head == null) {

return null;

}

Node current = head;

while (current.next != head) {

current = current.next;

}

return current;

}

public void display() {

if (head == null) {

System.out.println("List is empty.");

return;

}

Node current = head;

do {

System.out.print(current.data + " ");

current = current.next;

} while (current != head);

System.out.println();

}

public static void main(String[] args) {

CircularLinkedList list = new CircularLinkedList();

// Insert elements into the list

list.insert(10);

list.insert(20);

list.insert(30);

list.insert(40);

list.insert(50);

System.out.println("Original List:");

list.display();

int elementToInsert = 25;

list.insert(elementToInsert);

System.out.println("List after inserting " + elementToInsert + ":");

list.display();

}

}